AD A033102

(12)
B.S.

R-1872-PR
September 1976

AIDA: An Airbase Damage Assessment Model

D. E. Emerson

A report prepared for

UNITED STATES AIR FORCE PROJECT RAND

Rand
SANTA MONICA, CA. 90406

The research described in this report was sponsored by the United States Air Force under Contract No. F44620-73-C-0011 — Monitored by the Directorate of Planning, Programming and Analysis, Deputy Chief of Staff, Research and Development, Hq USAF.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. ~~REPORT NUMBER~~ 14 R-1872-PR	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) 6 AIDA: AN AIRBASE DAMAGE ASSESSMENT MODEL		5. ~~TYPE OF REPORT & PERIOD COVERED~~ 9 Interim Rept.
		6. ~~PERFORMING ORG. REPORT NUMBER~~
7. AUTHOR(s) 10 D. E. Emerson		8. ~~CONTRACT OR GRANT NUMBER(s)~~ 15 F44620-73-C-0011
9. PERFORMING ORGANIZATION NAME AND ADDRESS The Rand Corporation 1700 Main Street Santa Monica, Ca. 90406		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS FCRC Office (AF/RDXTR) Director of Planning, Programming & Analysis Hq USAF, Washington, D.C. 20330		12. ~~REPORT DATE~~ 11 Sep[redacted] 1976
		13. ~~NUMBER OF PAGES~~ 88
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) 12 95p.		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for Public Release; Distribution Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

No restrictions

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Damage Assessment
Military Air Facilities
Bombardment (Attack)
Targets
Computer Programs
AIDA

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

see reverse side

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

1B

Describes the airbase damage assessment (AIDA) computer model, designed for rapid examination of the results of conventional air attacks on complex targets. The model incorporates both a Monte Carlo and a deterministic mode of operation and has a variety of possible applications. It can be used to help plan effective attacks against complex target sets, or--by testing alternative attack headings, aim points, and the like--to examine the tradeoffs between damage to primary and secondary targets. It should also prove useful in airbase protection studies, both to provide realistic and detailed samples of possible damage patterns and to assess different options. Damage statistics for up to 250 individual targets can be quickly assessed for attacks involving as many as 50 delivery passes and 10 types of weapons. Both point-impact and area weapons can be handled, and targets may be grouped into 20 different vulnerability categories to distinguish different levels of weapon effectiveness. A complete user's guide and program listing are included. (PB)

R-1872-PR
September 1976

AIDA: An Airbase Damage Assessment Model

D. E. Emerson

A report prepared for

UNITED STATES AIR FORCE PROJECT RAND

Rand
SANTA MONICA, CA. 90406

PREFACE

This report describes the Airbase Damage Assessment (AIDA) computer model, designed for rapid examination of the results of conventional air attacks on complex targets. A user's guide and program listing are included.

AIDA was developed by Rand for use in a study, conducted at Hq USAFE, of conventional air attack effectiveness. The model incorporates both a Monte Carlo mode and a deterministic (or expected-value) mode of operation. Approximations used in the expected-value computations, while differing from those used by the Joint Munitions Effectiveness Manual (JMEM), are equivalent in precision to those of the JMEM hand methods. A large body of publications furnishes background for the methods used and for input values required by AIDA.

AIDA has a variety of possible applications. It can be used as an aid in planning effective attacks against complex target sets, or--by testing alternative attack headings, aim points, and the like--in examining the tradeoffs between damage to primary and secondary targets. It should also prove useful in target (e.g., airbase) protection studies, both to provide realistic and detailed samples of possible damage patterns and to assess different protection options.

AIDA has been discussed with, and made available to, the Operations, Intelligence, Plans, Logistics, Engineering, and Communications staffs at Hq USAFE; to the staff of the Assistant Chief of Staff for Studies and Analysis at Hq USAF; and to a number of other DoD and NATO organizations. This report is being published to provide a record of the model and to make it available to a wider audience. The computer program is available from The Rand Corporation.

This work was conducted under the Project RAND research project entitled "Rand Ramstein Activity."

SUMMARY

This report describes a new computer model that permits examination of conventional bombing attacks on complex targets--e.g., on an airfield. A complete user's guide is included. Damage statistics for up to 250 individual targets can be quickly assessed for attacks involving as many as 50 delivery passes and 10 types of weapons. Both point-impact weapons (such as general-purpose (GP) bombs and precision-guided munitions (PGMs)) and area weapons--cluster bomb units (CBUs)--can be handled, and targets may be grouped into 20 different vulnerability categories to distinguish different levels of weapon effectiveness. If the user is concerned only with the expected numbers of hits with point-impact weapons, and is not interested in either CBU weapons or the coverage and damage variations expected with point-impact weapons, a special, more efficient expected-value mode is provided.

In its basic mode, AIDA determines the actual impact points (pattern centroids for CBUs) by Monte Carlo procedures--i.e., by random selections from the appropriate error distributions. GP bombs and PGMs that impact within a specified distance of a target are classed as hits, and the results include the total number of hits on each target and the cumulative probability of kill. For CBU munitions the program assesses the fraction of each target covered by each pattern, and the results include the fractional coverage from all patterns and a cumulative probability of kill for each target. In addition to these results for the complete attack, the attack can be repeated automatically for several trials to provide statistics on the average damage levels to be expected. In the special expected-value mode, average hit densities are determined directly, without recourse to Monte Carlo procedures.

In the basic mode, up to 5 targets may be designated as runways or taxiways suitable for aircraft operations, and the model will examine these to see if an area of specified size is available for such operations; if not, the minimum number of craters that would need to be repaired to obtain an area of that size is determined.

The AIDA program is designed so that many cases can be examined conveniently during each run--a feature that will prove useful, for

example, when one desires to quickly examine several attack options. Computer output includes both the input data and the attack results that have been specified. Most model features are illustrated with a sample problem.

CONTENTS

I. MODEL DESCRIPTION

The AIDA (Airbase Damage Assessment) computer model permits examination of bombing attacks on a complex set of targets--e.g., on an airfield. In AIDA's basic mode of operation, the actual bomb impact points are obtained by Monte Carlo procedures,* and the attack can be repeated for several trials to provide statistical estimates of the average damage and variability of that damage for each of the many targets. Alternatively, an expected-value mode is offered when only point-impact weapons are employed in the attack and when the user is interested only in the expected numbers of hits. With a variant of the expected-value mode, one may also generate hit-density patterns for complex attacks without specifying an actual target system. Both modes of operation and several different sets of problems may be treated by successive cases during a single computer run. The several features available with AIDA are illustrated in Sec. IV.

In AIDA the target system may be composed of up to 250† separate targets, for example, shelters, hangers, maintenance buildings, runways, taxiways--even pipelines. The complete attack may consist of up to 50 distinct weapon-delivery passes. Each target is a rectangle of specific size and orientation and an attack pass is defined by the expected probability of arrival, a heading, and the aim point, delivery accuracy, and dispersion for a stick of weapons. Targets may be grouped into 10 or 20 different vulnerability categories and there may be up to 10 different kinds of weapons dropped in an attack.

This computer model has a variety of possible applications. Since an entire airbase may be represented rather accurately, AIDA can be used

---

*That is, the actual mean point of impact and the actual impact points are determined by random variates drawn from the appropriate error distributons.

†Most array dimensions may be changed fairly readily, and they have been in several instances. Some of the dimensions mentioned in this report differ from those outlined in an earlier version of AIDA; the changes were made to maximize program utility while limiting core storage requirements. Users will find instructions for changing program dimensions at the beginning of AIDA's MAIN subroutine.

to study how one might best design an attack to damage both primary and secondary targets. Whereas many bomb-damage aids deal only with a single target (or set of like targets) and provide no direct evidence as to the collateral damage to be expected, AIDA could be used to help design an attack that would get the maximum benefit from those bombs that miss the primary target. AIDA should also prove useful in a variety of studies of base protection and repair, in that it could provide a more realistic and detailed picture of possible damage than is customarily available. Yet another area in which AIDA may prove useful is as the first step in an analysis of the sortie-generation capabilities of a damaged base. Hopefully the results of this model can be used as input to a sortie-generation model,* and the combination may then be able to provide a much better understanding of the effects of attacks on aircraft operations than exists today. In addition to analyzing problems involving airfield complexes, AIDA could just as easily be used to assess attack options and expected damage levels for attacks against the complex target arrays found at field headquarters, SAM sites, supply depots, etc.

Although AIDA is straightforward and simple mathematically,† its substantial flexibility should prove useful in a wide range of studies. For the reader who wants more detail on the methodological treatment than is discussed in the body of this report, a program listing is included in Appendix C.

MONTE CARLO MODE

In this mode of AIDA, weapons may be of two basic types: point-impact weapons (such as general-purpose (GP) bombs and precision-guided munitions (PGMs) or area weapons (such as cluster bomb units (CBUs)). A weapon reliability may be associated with each kind of weapon. For each kind of point-impact weapon an effective miss distance (EMD) may be specified for each target type (i.e., that miss distance at which the weapon is effective and an impact is to be categorized as a hit).

---

*One that can account for the effects of various levels of damage to the various elements of a base.

†Approximations used in the AIDA computations, while different from those used in the Joint Munitions Effectiveness Manual (JMEM), are about equivalent in precision to those of the JMEM hand methods.

When this is done, target coverage is computed as that fraction of the target area that is covered by a circle having a radius of EMD and centered at the impact point. Additionally, the user may specify a value for target kill probability, given a hit (as defined above), or he may specify a different radius than the EMD and that will also be used for computing a value of coverage. This option may be specified individually for each target type and weapon type, as with the EMD. For example, with this feature one can use the EMD to represent the radius of the mean area of effectiveness (MAE) for severe structural damage to a building, and the alternate value to represent the radius of the MAE for severe damage to the building's contents, given a hit on the structure. The final results will include the cumulative coverage fractions for each target for all point-impact weapons, corresponding to both the EMD and to the optional factor, computed according to the relation:

$$FC = 1 - \prod_{\text{all hits}} (1 - C_{w,t}(i)) \ ,$$

where in the first case, $C_{w,t}(i)$ is the target coverage based on the EMD of the *ith* hit on the target of type t by a weapon of type w; in the second, it is assessed according to the user specified option.

If weapons are specified as CBUs, the model first computes the fraction of the area of each target that is covered by the rectangular bomblet pattern. The total fractional coverage of a target for all passes is that fraction of the area that has been covered by one or more patterns. If a probability of kill is associated with "coverage" by CBU patterns, the model will generate the total probability of kill, taking into account the actual position of each of the CBU patterns that covered any portion of the target, according to the relation:

$$PK = 1 - \sum_{i=1}^{T} \left\{ \prod_{a=1}^{M} \left(1 - p_{k_{w,t}} \right)^{N(a,i)} \right\} \Big/ T \ ,$$

where a = the attack number,

i = a point on the target grid,

$N(a,i)$ = the number of times point i was "covered" during attack a,

M = the total number of CBU attacks,

T = the total number of target grid positions, and

$p_{k_{w,t}}$ = the probability of kill of a portion of a target of type t that is "covered" by a bomblet pattern from a weapon of type w.

The results for each trial include the number of hits by point-impact weapons and the fractional coverage by CBUs for each target as well as the point-impact weapon coverage (FC) and CBU kill probability (PK); in addition, for the targets that the user has specified (for a maximum of 20 targets other than the runways and taxiways), the impact points and weapon types are printed for up to 25 weapons per target. For multiple trials, the results for each target include the fraction of trials with at least one hit, the average number of hits and average CBU coverage, the standard deviation of these two measures, and the average values of FC and PK for the several trials. A full description of the output options is presented in the next section and is summarized in Table 1, p. 9.

The user may also specify* that certain (up to 5) of the (rectangular) targets are actually runways or taxiways that are suitable for aircraft operations. The model will then test to see if such operations are possible from these areas; i.e., tests are made to see if the minimum clear length and width required for operations are available after the attack. In testing for runway availability, only point-impact weapons are considered, and the crater radius is taken as the EMD. Up to 250 hits can be stored and examined for each such target (the locations will be listed along with hit data on other targets). If the runway does not meet the minimum requirements, the user may request an assessment of how few craters would need to be repaired to meet those requirements. The user may also request an approximate computer plot of the impact points for each runway.

---

*By defining as type #1 targets.

Coverage by CBU patterns is not taken into account in the runway assessments;* examination of the effects of CBUs on the runway would require that the user either make a visual estimate using the approximate computer plot or plot the impact points and the resulting bomblet patterns more accurately on a plan of the airbase.

EXPECTED-VALUE MODE

This mode operates with target and attack descriptions that are identical to those used with the Monte Carlo mode of operation. However, only point-impact weapons may be included in the attack. Computationally this mode derives an average value of the hit density† for each target and for each attack. These are combined to provide the total expected number of hits for all the attacks. Although this mode provides no evidence regarding the statistical variations that must be expected in actual hit patterns, it does provide a quick, efficient means of assessing the expected values. Since, for many problems, a rather large number of Monte Carlo trials is required in order to get a reasonably accurate estimate of average values, one can sometimes use both the Monte Carlo and expected-value modes to advantage. The Monte Carlo mode can first be used with a limited number of trials to provide gross estimates of target coverage and of the variability of hits; the expected-value mode can then speedily provide a reasonably accurate estimate of the average number of hits. As will be illustrated

---

*Except that the centroids are shown on the computer plots when they fall within the area included in the plot.

†The point value of the hit density is first determined at several points on each target for the specified aiming errors and dispersion; these values are computed for each of several positions along the intended line of bomb impacts. The average hit density is taken as the arithmetic average of these point values for the several target points and the several positions along the bomb impact line. If the target dimensions are small (i.e., less than one-quarter the RMS of the range and deflection errors projected parallel to each target edge), only the target corners are used for averaging. For larger targets, a uniform grid of internal points is established with spacing no greater than one-quarter the RMS noted above. The expected number of hits is computed as the product of the average hit density and the area of the target (including a border as wide as the EMD).

in Sec. IV, this two-step calculation would require only two more cards than would either calculation alone.

A special feature of the expected-value mode permits the user to quickly generate a hit-density grid. This feature is controlled by special target cards that may be used either alone, or in conjunction with normal target cards. Each such card generates a 17 × 17 grid of hit-density values measured over a square of specified dimensions (the dimension should be a multiple of 16). If no dimension is specified, hit densities are provided at 250 ft intervals over a 4000 × 4000 foot square. The southwest corner for each grid is placed at that position specified on the special target card. Such cards are identified by specifying target type #21--an entry that acts as a control signal within AIDA.

AIDA OPERATION

AIDA offers several features designed to simplify its operation and to permit a series of cases to be analyzed during a single computer run. Most are illustrated in Sec. IV. The first feature permits a multi-aircraft attack against the same objective to be specified simply; when two or more attacks have common parameters (i.e., heading, desired mean point of impact (DMPI), CEP, dispersion, arrival probability), a single entry will generate the additional attacks. Other convenience features derive from use of the REDO card (see the next section). When this card is encountered it acts as a terminator card, terminating the input for one case and announcing that there will be a subsequent case. If desired, an entirely new set of data may be input following a REDO card. Alternatively, the subsequent case may be a modification of the preceding case; a few simple inputs permit the user a wide range of modification options. One may change AIDA's mode of operation or any of the control variables. In addition, targets may be added to the prior list; or some may be removed and then others added. The same can be done with attacks. For example, this feature could prove particularly useful in investigating a series of alternative attack options on a given airfield.

AIDA's features provide substantial flexibility. For example, since target location is not restricted, it is possible to have two identical targets at the same location; by assigning these targets different type numbers, one can assess results for two weapon effects. This would be useful, for example, in determining expected personnel and materiel losses (targets with different MAEs) in open parking areas. The kill probability options available with impact weapons complement this technique. Consider, for example, two identically located buildings (of different target types), each with EMDs fixed by the MAE for a structural kill. If one also inputs an effective kill radius based on equipment damage for one of the buildings and the kill radius for personnel with the other, the Monte Carlo results will include estimates of the structural damage, as well as of the materiel damage and personnel losses resulting from a structural hit.

II. INPUT PROCEDURES

Up to seven basic types of cards can be used in operating AIDA, although only three are required. Four types describe the target and attack characteristics and three others are used to control operations. For each basic card the type is identified in the first four columns (left-adjusted):

TGT	target data card
ATT	attacker data card
ATT2	alternate attacker data card (optional)
EMD	weapon card (optional)
CONT	control card (optional)
REDO	controls sequential cases (optional)
END	last card

For two of these basic card types--ATT2 and EMD--supplementary cards are used. For the ATT2 card a following card (with additional data, as will be described) is mandatory. One to three supplementary cards may be employed with the EMD card, as will be described.

There may be as many as 250 TGT cards, 50 ATT or ATT2 cards, and 10 EMD cards. For a given case there may be at most 1 CONT card. The order of the cards is immaterial* except that a REDO card or an END card must be used to signify the completion of input for a given case. The targets and attackers are numbered, internally, in the sequence in which their descriptions are read in; each target may also have an alphanumeric designator (e.g., building number). A detailed description of how data are to be entered on each type of card is presented in Appendix A. The input data are printed for each case; Table 1 outlines the output options for the results. The program will function in its simplest form with only TGT cards, ATT or ATT2 cards, and an END card.

---

*The only exception is that if the special "target type #21" cards are used in conjunction with the special hit-density grid feature, one of these cards must be the last target card.

Table 1

OUTPUT CONTROL

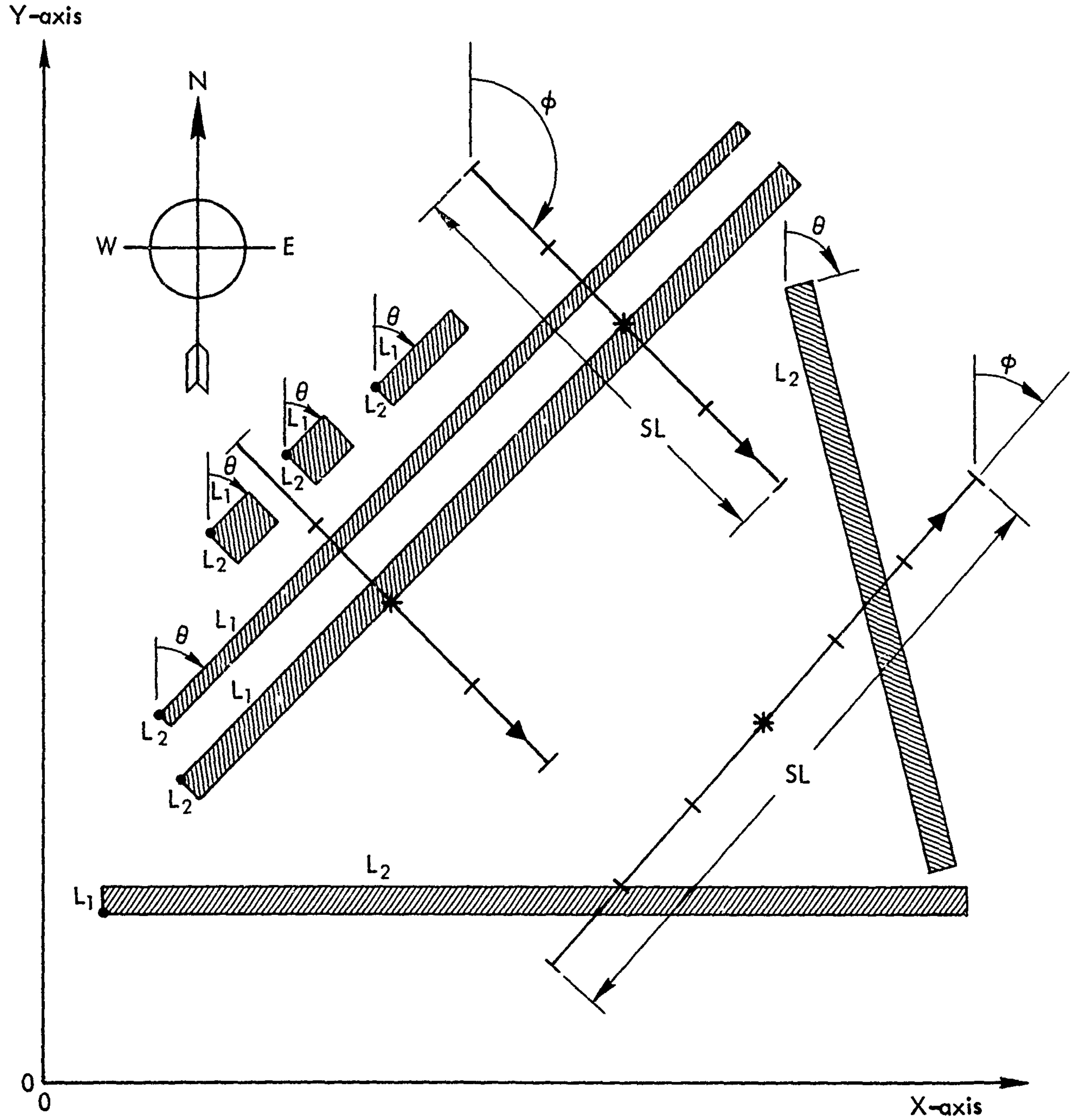
NPRINT[a] Control Value	For Each Trial: All Impact Points	All Hits (and Target Corners)	Stored Hit Data	Hit Summary	Runway Results	Multiple Trial Statistics
-2	X	X	X	X	X	X
-1		X	X	X	X	X
0			X	X	X	X
1				X	X	X
2					X	X
3						X
4					X[b]	X
5				X[c]		X

[a]Enter in Columns 23, 24 of CONT card.

[b]Compact listing of hits and required repairs for runways/taxiways.

[c]Compact listing of hits on each target.

A control (CONT) card will be required if advantage is to be taken of more than the most basic of AIDA's features; without this card AIDA examines only one Monte Carlo sample of the attack and provides the actual numbers of hits on all targets and the stored hit locations for specified targets. Specifically, a CONT card is needed if (1) more than one trial is required, (2) an alternative output mode is desired, (3) a different mode of operation is desired (e.g., the expected-value mode), or (4) the runway availability features are to be exercised. As explained more fully on pp. 38-39, this card is used to specify the number of trials, the mode of operations, the output formats, the dimensions of the minimum runway surface, whether or not the user wants the minimum repair requirement assessed, and the distances that the "minimum runway rectangle" will be shifted laterally and longitudinally in testing for an available area.



TARGETS

- • Reference (westernmost) corner
- L_1 Northeasterly heading boundary
- L_2 Southeasterly heading boundary
- θ Orientation angle

ATTACK PASSES

- ✱ DMPI
- ϕ Attack heading
- SL Stick length
- ⊢+ Nominal bomb impacts

Fig. 1 — Target and attack layout for AIDA

Several special features should be noted with respect to the EMD cards. To begin with, these cards are optional; if the EMD card is not used for one or more weapon types, these weapons are assumed to be point-impact weapons that must impact within the target outline to be counted as "hits." If more of the features are to be used for a particular type of weapon, an EMD card is input with the number of the weapon entered in Columns 11 and 12. If weapon reliability is less than unity, a 1 is placed in Column 6 of the EMD card and the reliability is entered in Columns 7 to 12 of a special card placed immediately following the EMD card. If the user wants to consider 20 types of targets, rather than only 10, a 1 is placed in Column 5, and data for the second set of 10 target types are entered on a supplementary card (2 cards if a 1 also appears in Column 6). The other entries for the EMD card and the special following cards differ, depending on whether it is a CBU-type munition or a point-impact weapon.

CBU munitions are denoted by a negative number in Columns 13 to 18 on the EMD card. The absolute value of that number is taken as the CBU pattern dimension along the flight direction; pattern width is specified in Columns 19 to 24. If the user wants to associate a probability of kill with CBU coverage on some or all of the target types, that fact is also denoted by the integer 1 in Column 6 of the EMD card. When this is done the program interprets entries on the following card (also used for weapon reliability) as p_ks for that weapon against the desired target types.*

Other special features are available for use with point-impact weapons. If a weapon can effectively damage a target when it actually falls near but outside the target outline, the EMD for a "hit" can be entered in Columns 13 through 72 (in 10 fields of 6 columns) of the EMD card (and auxiliary card) for the 10 (or 20) target types. The appropriate entry in many instances would be the radius of a circle whose area is equal to the MAE (as presented in JMEM manuals) for the corresponding target-weapon combination. In the case of hits on runways or taxiways, however, the appropriate entry is crater radius; when AIDA

---

*In ten 6-column fields from Columns 13 to 72.

checks for the availability of a minimum runway, each reliable impact is assumed to have a crater radius equal to EMD.

For each "hit" with a point-impact weapon an estimate is made of the fraction of the target area that is covered by a circle of radius EMD.* The user may also select one of the additional options (see Fig. 2); using the supplementary card he either may input a p_k for a particular weapon-target type or specify a radius different than EMD for defining another "effects" circle to be used in computing an additional coverage measure. In either case (p_k or radius) the value† is entered on the supplementary card in the field corresponding to the appropriate target type.

A REDO card terminates the input for one case and initiates a new case. If no additional entries are made on the REDO card, the targets and attacks for the next case will include all those entered in the preceding case. If some, but not all, of the preceding targets and/or attacks are to be included, the number of targets that are to be retained is entered in Columns 7 to 12 and the number of attacks to be retained is entered in Columns 13 to 18. The numbers to be retained are selected from the beginning of the ordered lists generated in the input process. If a negative entry (e.g., -1) is made in either or both of the target or attack fields, none of the prior targets and/or attacks will be considered in the new case.

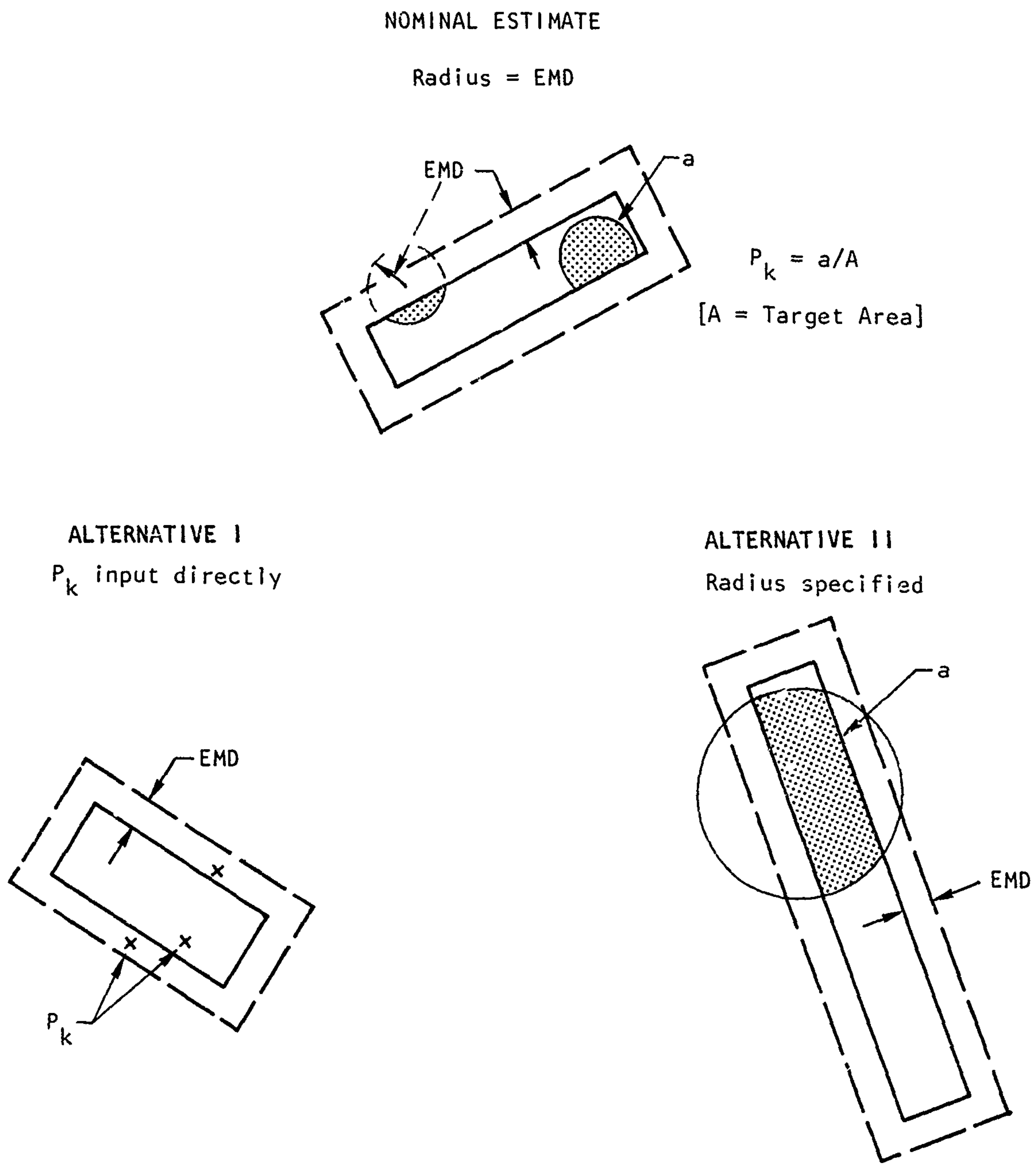
After specifying which--if any--of the prior inputs are to be retained, additional targets and attacks may be added with appropriate TGT and ATT cards. If a 1 is entered in Column 24 of the REDO card, the target list and/or the attack and weapons lists will not be included in the input listing, *if* no change has been made in those lists from the prior use.

Control conditions will remain as in the preceding case unless a new CONT card is entered with appropriate data. An END card must be used, at the end of the last case, to complete the card deck.

---

*As noted earlier, the cumulative coverage of a target is given by 1 minus the product of the probability of noncoverage to all hits; this is equivalent to saying that in estimating the target area covered by a particular weapon the effect of prior hits is neglected.

†A value greater than 1 is interpreted as a radius.



CUMULATIVE COVERAGE

$$FC = 1 - \prod_{i\,=\,\text{all hits}} (1 - P_{k_i})$$

Fig. 2 — Point-impact weapon coverage estimation options

III. PROGRAM NOTES

AIDA is written in FORTRAN IV and first operated on the Honeywell 6060 used with Hq USAFE's Worldwide Military Command and Control System (WWMCCS); subsequently, it has been run on the IBM 360/370 series. AIDA is easily transferable to other computers because there are no system-unique features employed.* The program is composed of approximately 1950 card images organized into a MAIN routine and 17 subroutines. As presently dimensioned, AIDA operation requires 36K words of core. The organization of the various subroutines and their functions are indicated in Fig. 3; this structural outline of AIDA should also prove useful for anyone who wants to overlay the program so as to reduce core storage requirements. Potential users will find that the source code is moderately well annotated with comment cards. A full listing of the definitions for the key variables and arrays will be found in Appendix B.

AIDA's CPU requirements obviously can vary widely depending on the complexity of the problem (i.e., numbers of targets and attacks and use of the special runway features) and upon the required number of replications. For problems involving about 150 targets, 750 bombs, and 20 replications of the attack, computing times have run between 1-1/2 to 2 minutes on the Honeywell machine. To limit these requirements the process used to test for hits with point-impact weapons has been designed to reduce the TARGETS X BOMBS dimensionality problem somewhat, by ordering targets in a particular manner and by only checking those targets "near" each bomb impact point. Nevertheless there is a very substantial amount of checking and computation required.

The computation problem is especially severe when CBU munitions are used. To estimate the joint coverage of all CBU patterns, the targets each have a uniform grid of points superimposed, and each of those points is checked to see if it lies within one or more of the patterns.

---

*Minor format changes may be required in a few instances; name changes will probably be necessary for the random number generator.

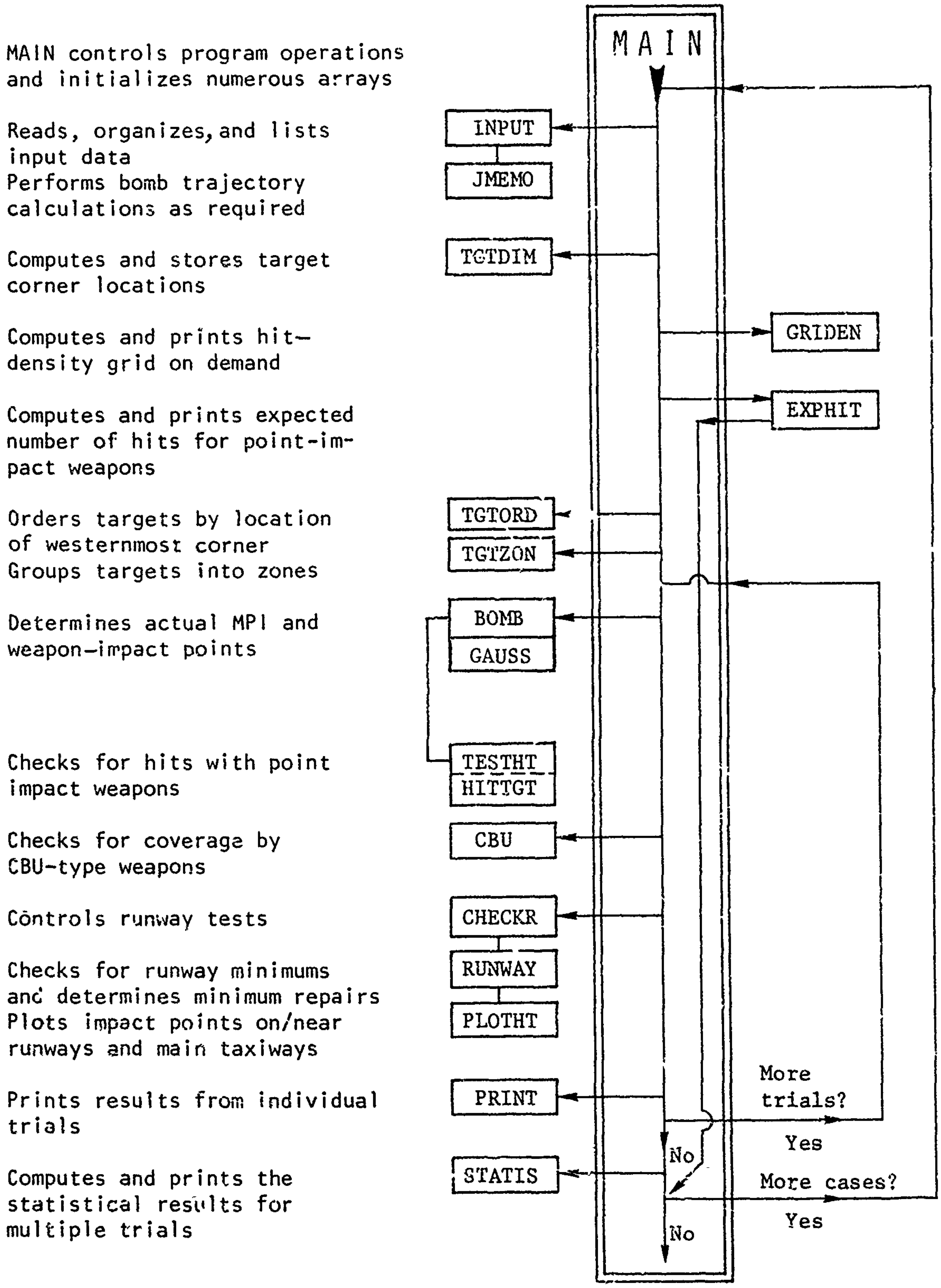


Fig.3— Organization and function of the AIDA subroutines

The fraction of the target grid points that is covered by one or more CBU patterns is used as the estimate of fractional coverage. For small targets only a 4 × 4 grid is used, but for the very large targets (dimensions in excess of 1000 × 1000) a 16 × 16 grid is used;* quite obviously the computational requirements are affected by the presence of large targets and the use of CBUs, but not, fortunately, in direct proportion to the number of grid points.

The procedure used for checking the availability of an adequate section of runway also involves substantial processing. A rectangle of the required dimensions is positioned first at a corner of the runway and tested to see if there is a hit within the rectangle. If there is, the rectangle is moved 5 ft† laterally and rechecked; this is repeated across the runway or until an open area has been found. If none is found the rectangle is shifted 250 ft† along the runway and the process is repeated. If the dimensions of the minimum usable runway section are small, or if the runway is large and/or there are many hits, a very substantial amount of processing can be required. These requirements are further increased when an assessment of the minimum repair requirements is requested. Furthermore, if there are other runways, or taxiways suitable for emergency flight operations (signified by entry as a target type #1), the entire process must be repeated for each surface (for a maximum of 5). Whenever the problem can be effectively reduced to one of establishing the availability of the minimum clear width (e.g., when a 9000 ft runway is being cut at the center to deny a 6000 ft clear length), the minimum clear length should be set equal to the runway length to avoid going through the lateral shift sequence unnecessarily (11 times in the example).

---

*It will be noted that hits may not be recorded when a target is very large and the CBU pattern so small, proportionately, that the pattern can fall between grid points.

†These are the default values; the user may specify a different value.

IV. SAMPLE PROBLEM

The layout of Base XYZ is shown in Fig. 4. This base consists of a 150 ft × 8000 ft main runway, several taxiways, a parking ramp, eight support facilities, and a housing area. To examine the effectiveness of bombing attacks against this target complex with AIDA, one first must describe the target elements and the attack in a common coordinate system. Targets are defined by their westernmost corner, their size, and their orientation; the attack heading and the desired mean point of impact fix the attacks. For this illustration four medium bombers will each drop 25 bombs in an effort to cut the runway at two points; two others are targeted on the operations building near the main taxiway, and one will aim at the electronics shop. In addition, one fighter-bomber is assigned to dive-bomb each of the main aircraft maintenance buildings, B1 and B2, and one will drop a stick of 5 CBUs on the housing area.

INPUT

Figures 5a and 5b reproduce the card images needed to describe this sample problem and to control four distinctly different assessments. These assessments are designated as "cases" and illustrate the following:

Case 1. Hit statistics for all targets, using 5 trials.
Case 2. Detailed results of a single trial.
Case 3. Results using the expected-value mode, with a sample hit-density grid.
Case 4. Summary results of the runway attack, using 25 trials.

For clarity, the control (CONT) card, the target (TGT) cards, the weapon (EMD) cards, and the attack (ATT and ATT2) cards are listed in order; the ordering of cards for a given case is actually immaterial except that the EMD card pairs and the ATT2 card pairs must be together. A REDO card or an END card defines the end of the input data for a case.

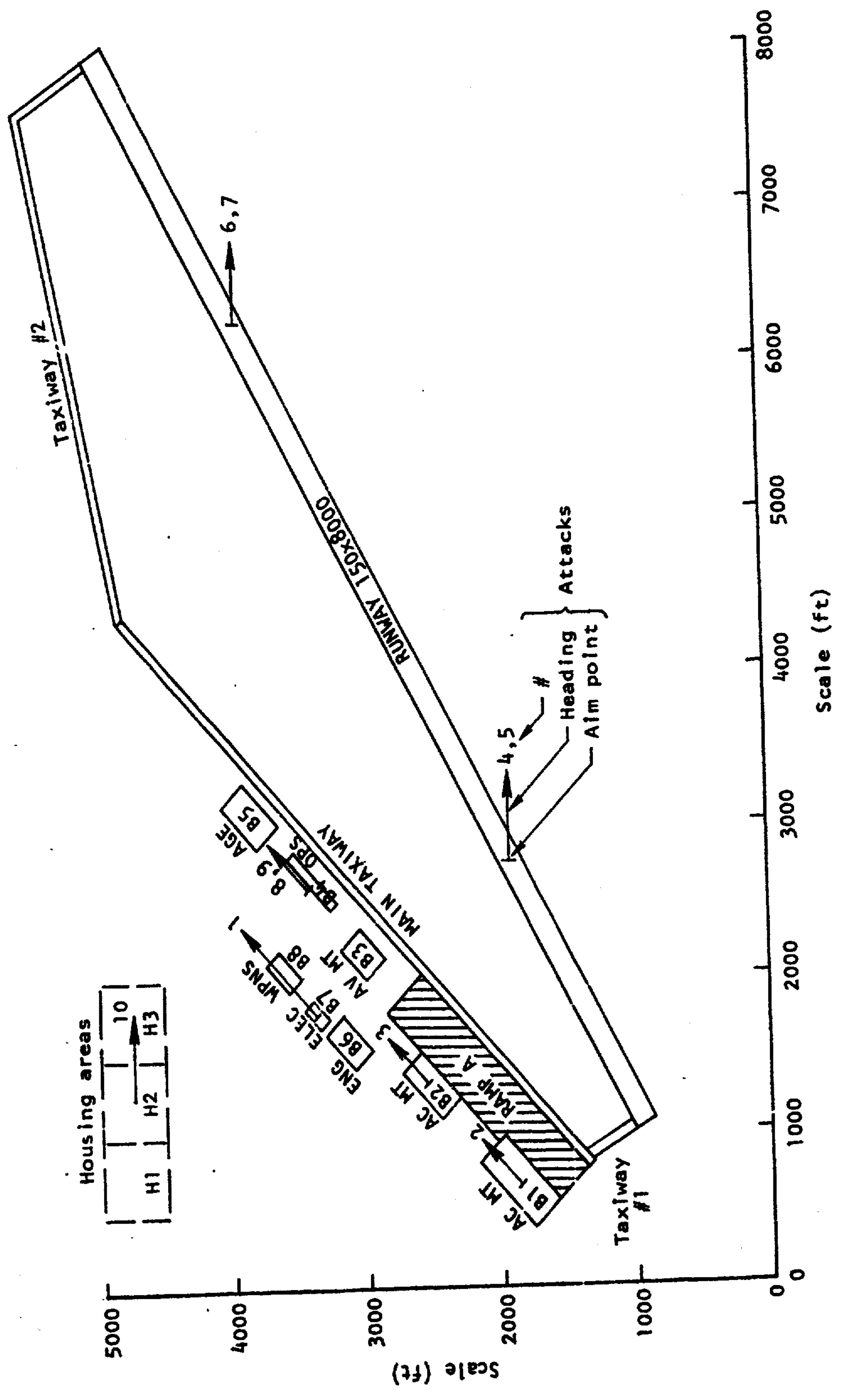


Fig. 4 — Base XYZ

```
         ┌Monte Carlo mode with fixed seed for random number generator
         │  ┌Number of trials
         │  │  ┌Statistical summary only
CONT     0  5  3                                        BASE XYZ

TGT   1000  1000  8000  150   60   1                    FUNWAY
           ┌Target coordinates
TGT    775  1350  5000   50   45   1                    MAIN TWY

TGT    540  1610  1750  175   45   3                    RAMP A
                   ┌Target dimensions
TGT    400  1325    50  375   60   2                    TXWY #1

TGT   4400  4800  1300   50   78   2                    TXWY #2   ┌Optional
                             ┌Target orientation                   descriptors
TGT    400  1750   600  250   45   4   1                AC MT B1

TGT   1100  2450   400  200   45   4   1                AC MT B2
                                   ┌Target type
TGT   2000  3000   250  150   45   5                    AV MT B3

TGT   2450  3220   500   60   45   4                    OPS   B4

TGT   2850  3800   400  250   45   5                    AGE   B5

TGT   1425  3075   300  175   45   4   1                ENG   B6

TGT   1689  3325   150   80   45   5     Stores hit locations ELEC B7

TGT   1930  3575   225  100   45   5   1                WPNS  B8

TGT    500  4500   500  500    0  10                     H 1

TGT   1000  4500   500  500    0  10                     H 2   Housing areas

TGT   1500  4500   500  500    0  10                     H 3
                         ┌EMD--Hits are recorded for impacts
                         │     within EMD of the target
EMD  1   1    22    22    22   40   40
      ┌Weapon reliability    ┌RAD2--Damage to type #4 and #5 targets
     .95                   50   75   is assessed with these radii, given a hit
   ┌Indicates that card follows
EMD  1   2    28    28    28   50   50
          └Weapon type
     .95                       60   90
          ┌Signifies CBU weapon                     ┌ P_k for type #10 targets
EMD  1   3   500   250                              │ is assessed assuming
             └Pattern size, 500 x 250               │ 25% chance of kill
     .90                                       .25    within area covered
```

TARGET CARDS

WEAPON CARDS

Fig. 5a — Input control card deck, Case 1

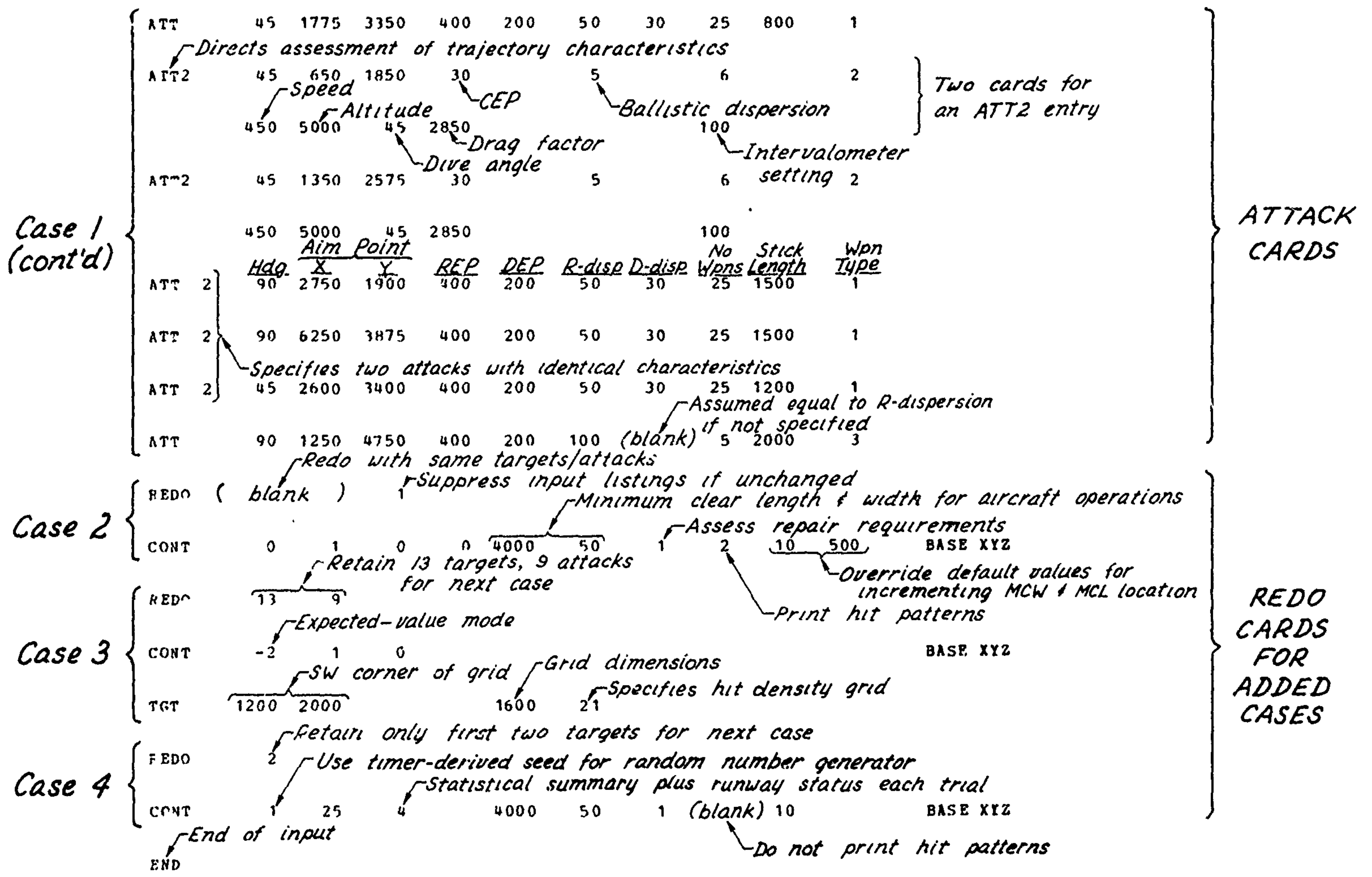


Fig. 5b — Input control card deck, Cases 1-4 (cont'd)

A careful review of the various annotations in Figs. 5a and 5b along with a reading of Appendix A, hopefully will lead to a full and rapid understanding of the various input requirements and optional features provided by AIDA. As can be seen in Case 1, the first card directs that the Monte Carlo mode be used and that a statistical summary of five replications of this attack be printed; no assessment of runway availability is requested for this case. The specifications for the targets are straightforward. The weapon data are somewhat more complex; two cards are used in describing each weapon's effectiveness against the 10 types of target (four cards would be required for 11 to 20 target types). The first card lists the EMD for 10 target types and the second card lists weapon reliability and the optional effectiveness descriptor (an alternate radius, or a p_k). In this example weapon type #1 has an effective (crater) radius against paved surfaces (target types #1, #2, and #3) of 22 ft; against target types #4 and #5 a hit will be assessed if the impact is within 40 ft of these structures. The fractional damage on these structures, given a hit, is to be based on a 50 ft effects radius for target type #4, and a 75 ft radius for target type #5. The third weapon type is a CBU and the special input procedures are illustrated.

The 10 attacks are listed last. The first (and fourth through tenth) attack uses the simpler ATT specification; the REP, DEP, dispersion, and stick length are given directly in feet, measured in the ground plane. For the second and third attacks--the fighter-bombers directed to attack the main maintenance hangars B1 and B2--these factors are to be computed, based on the aircraft's attack characteristics; both will release their last bomb at a 5000 ft altitude in a 45 deg dive at 450 kn with an intervalometer setting of 100 ms. Aiming accuracy and dispersion in the plane normal to the trajectory are 30 mils and 5 mils, respectively. For three of the attacks, two aircraft will attempt to fly the identical path, so only one card is needed to describe each pair.

The remaining eight cards are all that are needed to define and control three additional cases. Case 2 simply calls for a single Monte Carlo attack, but with full printout and with an examination of the

availability of an undamaged 50 ft × 4000 ft section of either the runway or main taxiway for aircraft operations.

Case 3 requests an estimate of the expected numbers of hits using the expected-value mode; since this mode will not function for CBU-type munitions, only the first nine attacks are retained (the housing area targets for the tenth attack are also dropped). The target card that is added in this case directs the additional computation of a 1600 ft × 1600 ft hit-density grid, located as noted.

Case 4 focuses on the availability of a minimum surface for aircraft operations. Only the runway and main taxiway are retained as targets; however, all attacks are to be considered, since any crater must be taken into account, whether it represents a hit on the intended target, or collateral damage from some different attack. Twenty-five attack trials are to be run with the Monte Carlo mode, repair requirements are to be assessed, and the trial-to-trial runway results are to be printed along with the statistical summary for the 25 trials.

OUTPUT

The very first output (Fig. 6), presented even before the main summary of the input data, lists the input/output for any trajectory calculations that were required; in this case this was necessary for attacks 2 and 3. The basic input listing follows; as will be noted, the factors determined with the trajectory calculations are listed in their normal location in this format. It will also be noted that the targets and the attacks were assigned numbers in the order in which they were located in the input deck; the pairs of identical attacks each were assigned two numbers.

The target damage statistics for five Monte Carlo replications of the attack in Case 1 are shown in Fig. 7. The various annotations will help to clarify the nature of the various statistics that are provided. As can be noted in conjunction with the type #10 targets, "hits" are not assessed with CBU weapons; the results are in terms of coverage and fractional kill (PK). Perhaps the most significant--and typical--observation that might be drawn from these sample statistics is the very substantial uncertainty that must be attached to any measures of

JMEM FORMAT ATTACK DATA

Inputs and trajectory data for dive bombing attacks on buildings B1 and B2

ATT NO	HDG / SPEED	X-MPI / ALTITUDE	Y-MPI / DIVE	CEP (FEET) / TEMP1	(DEP) / TEMP2	DISP / WIND	TD/HP	T	TD
2	45.	650.	1850.	30.	0.	5.			
	450.	5000.	45.	2850.	0.	0.	0.0	0.100	0.0
			TF 7.83 SEC	SR	6795. FEET	IMPACT ANGLE 55.85 DEG			(FUZING ANGLE 0.0 DEG)
1	45.	1350.	2575.	30.	0.	5.			
	450.	5000.	45.	2850.	0.	0.	0.0	0.100	0.0
			TF 7.83 SEC	SR	6795. FEET	IMPACT ANGLE 55.85 DEG			(FUZING ANGLE 0.0 DEG)

Intervalometer setting

```
****************************************************************************************************
                    *****AIDA*****     AN AIRBASE DAMAGE ASSESSMENT MODEL
                    DEVELOPED BY THE RAMSTEIN OFFICE OF THE RAND CORPORATION
****************************************************************************************************
NO OF TRIALS   5 NPRINT  3  MODE  0   MCL     0( 250)   MCW    0( 5)   MIN REPAIR 0  PLOT HITS 0   TEST  0
****************************************************************************************************
```

Statistical summary only

***** BASE COMPLEX NAME - BASE XYZ *****

TARGET DATA

NUMBER	X-DIM	Y-DIM	ME LIMS	SF LIMS	ANGLE	TGT TYPE	STOPE	BLDG NO
1	1000.	1000.	8000.	150.	60.	1.	0.	RUNWAY
2	775.	1350.	5000.	50.	45.	1.	0.	MAIN TWY
3	540.	1610.	1750.	375.	45.	3.	0.	RAMP A
4	800.	1325.	50.	175.	60.	2.	0.	TXWY #1
5	4400.	4800.	3300.	50.	78.	2.	0.	TXWY #2
6	400.	1750.	600.	250.	45.	4.	1.	AC MT B1
7	1100.	2450.	400.	220.	45.	4.	1.	AC MT B2
8	2000.	3000.	250.	150.	45.	5.	0.	AV MT B3
9	2450.	3220.	500.	60.	45.	4.	0.	OPS B4
10	2650.	2800.	400.	250.	45.	5.	0.	AGE B5
11	1425.	3075.	300.	175.	45.	4.	1.	ENG B6
12	1680.	2325.	150.	80.	45.	5.	0.	ELEC B7
13	1930.	3575.	225.	100.	45.	5.	1.	WPNS B8
14	500.	4500.	500.	500.	0.	10.	0.	H 1
15	1000.	4500.	500.	500.	0.	10.	0.	H 2
16	1500.	4500.	500.	500.	0.	10.	0.	H 3

ATTACK DATA

As computed with the JMEM method

NUMBER	HDG	X-DMPI	Y-DMPI	REP	DEP	P-DISP	D-DISP	NO WPNS	LENGTH	WPN TYPE	ARRIVAL
1	45.	1775.	3350.	400.	200.	50.	30.	25.	800.	1.	1.000
2	45.	650.	1850.	151.	117.	28.	23.	6.	86.	2.	1.000
3	45.	1350.	2575.	151.	117.	28.	23.	6.	86.	2.	1.000
4	90.	2750.	1900.	400.	200.	50.	30.	25.	1500.	1.	1.000
5	60.	2750.	1900.	400.	200.	50.	30.	25.	1500.	1.	1.000
6	90.	6250.	3875.	400.	200.	50.	30.	25.	1500.	1.	1.000
7	90.	6250.	3875.	400.	200.	50.	30.	25.	1500.	1.	1.000
8	45.	2600.	3400.	400.	200.	50.	30.	25.	1200.	1.	1.000
9	45.	2600.	3400.	400.	200.	50.	30.	25.	1200.	1.	1.000
10	90.	1250.	4750.	400.	200.	100.	100.	5.	2000.	3.	1.000

CBU; see below

MISS DISTANCES ALLOWED FOR EFFECTIVE HITS

Crater radii vs pavements — EMD — Damage radii given a hit

WPN TYPE	WPN REL	TARGET TYPES 1	2	3	4	5	6	7	8	9	10
1	0.950	22.	22.	22.	40.	40.	0.	0.	0.	0.	0.
		0.0	0.0	0.0	50.000	75.000	0.0	0.0	0.0	0.0	0.0
2	0.950	28.	28.	28.	50.	50.	0.	0.	0.	0.	0.
		0.0	0.0	0.0	60.000	90.000	0.0	0.0	0.0	0.0	0.0
3	0.900	-500.	250.	0.	0.	0.	0.	0.	0.	0.	0.
		0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.250

CBU — Pattern size

Fig. 6 — Target, attack, and trajectory data, Cases 1-4

TARGET DAMAGE STATISTICS FOR 5 TRIALS

TARGET NUMBER	PERCENT ATTACKS HIT	AVERAGE HITS PER ATTACK	STD. DEV. OF HITS	AVG. CBU COVERAGE	STD. DEV. COVERAGE	AVG. BOMB COVERAGE EMD	AVG. BOMB COVERAGE OTHER	CBU PK	BLDG NO.
TARGET TYPE # 1									
1	100.0	17.20	3.49	0.0	0.0	0.018	0.0	0.0	RUNWAY
2	60.0	4.80	5.63	0.0	0.0	0.014	0.0	0.0	MAIN TWY
		22.00		0.0					
TARGET TYPE # 2									
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	TXWY #1
5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	TXWY #2
		0.0		0.0					
TARGET TYPE # 3									
3	100.0	7.00	4.30	0.0	0.0	0.019	0.0	0.0	RAMP A
		7.00		0.0					
TARGET TYPE # 4									
6	80.0	2.80	2.39	0.0	0.0	0.068	0.098	0.0	AC MT B1
7	80.0	5.60	4.34	0.0	0.0	0.289	0.380	0.0	AC MT B2
9	40.0	2.00	3.46	0.0	0.0	0.091	0.134	0.0	OPS B4
11	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	ENG B6
		10.40		0.0					
TARGET TYPE # 5									
8	20.0	1.20	2.68	0.0	0.0	0.057	0.177	0.0	AV MT B3
10	60.0	2.40	3.36	0.0	0.0	0.069	0.193	0.0	AGE B5
12	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	ELEC B7
13	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	WPNS B8
		3.60		0.0					
TARGET TYPE # 10									
14	0.0	0.0	0.0	0.06	0.08	0.0	0.0	0.014	H 1
15	0.0	0.0	0.0	0.27	0.26	0.0	0.0	0.079	H 2
16	0.0	0.0	0.0	0.18	0.19	0.0	0.0	0.045	H 3
		0.0		0.51					

Totals

Ramp A hit on 5 out of 5 trials

Average fraction of building covered by damage effects with radius = EMD (40 ft)

Average fraction of building covered by damage effects with 50 ft radius, given an impact within EMD feet

Note highly variable results; standard deviation greater than average

6 to 27% of the housing areas covered, on the average

1 to 8% of the areas are killed, on the average

Percentage of targets in category that receive at least one hit, averaged over the trials

DAMAGE STATISTICS BY TARGET TYPE

TARGET TYPE	AVERAGE PERCENT HIT	STANDARD DEVIATION	COVERAGE EMD	COVERAGE OTHER	CBU
1	80.0	27.4	1.7	0.0	0.0
2	0.0	0.0	0.0	0.0	0.0
3	100.0	0.0	1.9	0.0	0.0
4	50.0	17.7	11.6	15.7	0.0
5	20.0	11.2	6.1	15.1	0.0
10	0.0	0.0	0.0	0.0	4.6

Percent of total target area (by type) that is covered by damage effects with radius = EMD, averaged over 5 trials

Same but using alternate effects radii; or, percent of target type killed

Percent of total target type area killed by CBU patterns

Fig. 7 — Target damage statistics, Case 1

the "average level of damage": as can be seen, the standard deviation is frequently as large, or larger, than the mean.

Case 2 called for a full printout of one trial, but with the input data suppressed. These results are shown in Figs. 8 through 10. The first results presented are the hit patterns on the runway (Fig. 8) and main taxiway (Fig. 9) as well as statements as to their status. As will be noted, all hits within an EMD of the target are included, since a crater will affect the surface up to a distance equal to the crater radius. In the case of the taxiway against which no attack was specifically planned, it is apparent that either an attack aimed at the operations building or one of those aimed at buildings B1 or B2 was wide and to the right of the aim point. A careful study of the target hit summary (Fig. 10) indicates that it was one of the attacks on the operations building, since the hits are with type #1 weapons. Using the EMD of 40 ft for structural damage and a 75 ft radius for the contents of type #5 buildings, the results show that this attack will destroy an estimated 15.7 percent of the structure and damage contents on 57.5 percent of the floor space of the avionics maintenance facility (B3); on the other hand, the AGE facility (B5), which is similarly located with respect to the intended aim point near the operations building, was missed entirely as was the intended target. One of the main aircraft maintenance hangers (B2) was hit. Other dive-bombing attacks were both wide and to the right, placing bombs on the main parking ramp. For the housing areas, there was no damage to the exposed targets in the three areas in this particular trial. The actual hit locations on building B2 and on the operating surfaces are shown at the bottom of Fig. 10. Of the four targets for which hits were to be retained (#6, #7, #11, and #13), only #7 was hit in this case.

Case 3 used the expected-value mode to provide estimates of the expected numbers of hits and to compute a hit-density grid. The latter is controlled by specifying a "type #21" target; as will be noted in Fig. 11, the grid dimension is printed in the position normally reserved for target angle. Any grids that have been specified are computed first and the sample requested in this case is shown in Fig. 12; as will be noted, the results are presented in terms of the expected numbers of

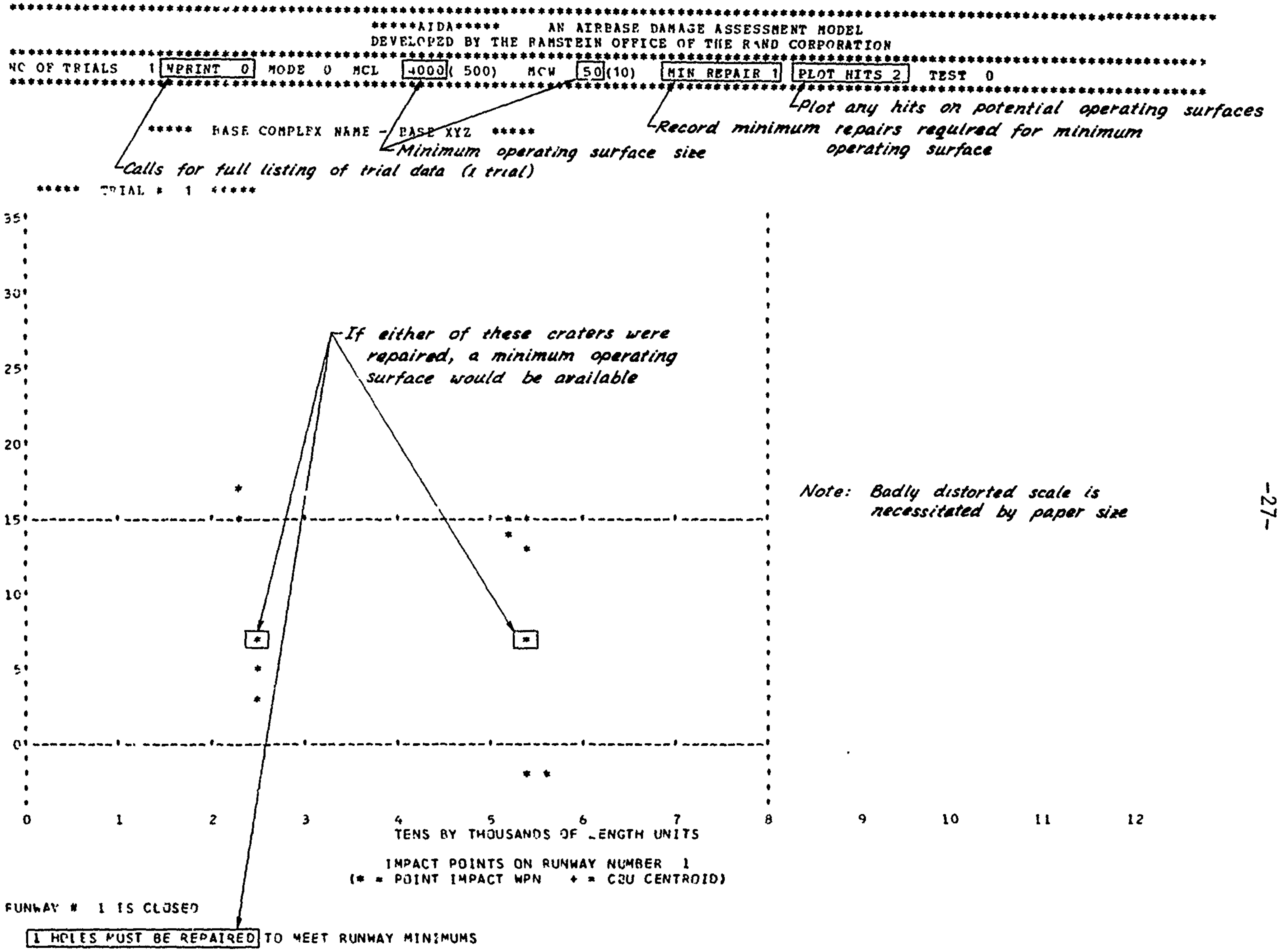
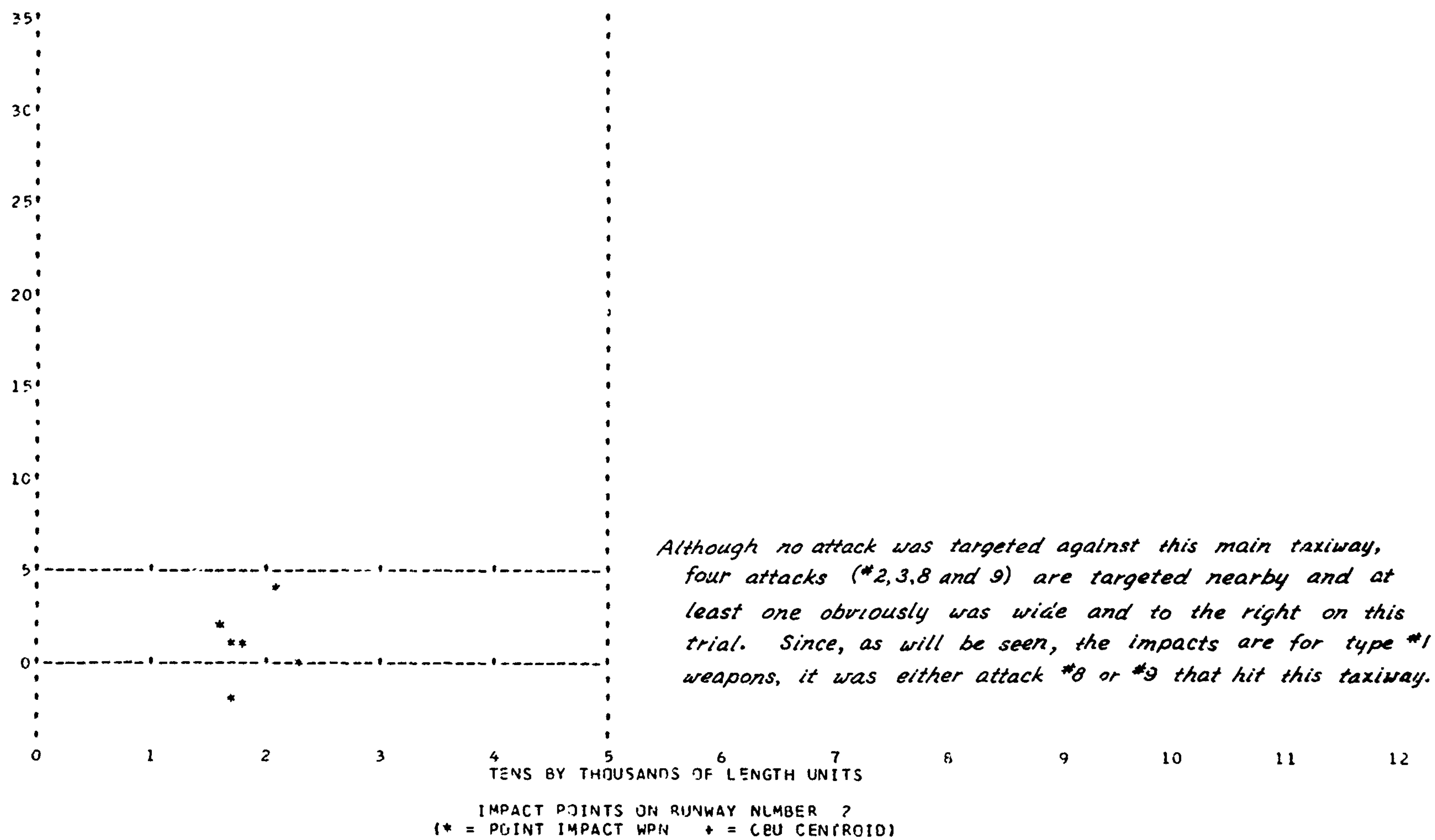


Fig. 8 — Runway hit patterns, Case 2



RUNWAY # 2 IS CLOSED

6 HOLES MUST BE REPAIRED TO MEET RUNWAY MINIMUMS

Fig. 9 — Taxiway hit patterns, Case 2

TARGET HIT SUMMARY

TGT NO.	NO. HITS	CHU COVERAGE	BOMBS EMP	BOMBS OTHER	CBU PK	BLDG NO.
		See sketches (Fig 9) and location listing below				
		** TARGET TYPE # 1 **				
1	$\boxed{13}$	0.0	0.0	0.0	0.0	RUNWAY
2	$\boxed{6}$	0.0	0.0	0.0	0.0	MAIN TWY
		** TARGET TYPE # 2 **				
4	0	0.0	0.0	0.0	0.0	TXWY #1
5	0	0.0	0.0	0.0	0.0	TXWY #2
		** TARGET TYPE # 3 **				
3	18	0.0	0.040	0.0	0.0	RAMP A
		** TARGET TYPE # 4 **				
6	0	0.0	0.0	0.0	0.0	AC MT B1
7	4	0.0	0.093	0.137	0.0	AC MT B2
9	0	0.0	0.0	0.0	0.0	OPS B4
11	0	0.0	0.0	0.0	0.0	ENG B6
		** TARGET TYPE # 5 **	With 4 hits, 15.7% of the area of this building covered by damage effects circles of radius = EMD (4)			
8	4	0.0	$\boxed{0.157}$	$\boxed{0.575}$	0.0	AV MT B3
10	0	0.0	0.0	0.0	0.0	AGE B5
12	0	0.0	0.0	0.0	0.0	ELEC B7
13	0	0.0	0.0	0.0	0.0	WPNS B8
		** TARGET TYPE # 10 **		57.5% of the building's area is covered by the alternate damage effects radius (75 ft)		
14	0	0.0	0.0	0.0	0.0	H 1
15	0	0.0	0.0	0.0	0.0	H 2
16	0	0.0	0.0	0.0	0.0	H 3

HIT LOCATION AND WPN TYPE FOR SELECTED TARGETS

TARGET NUMBER 7	X-DIM	Y-DIM	WPN TYPE
	1517.	2587.	2
	1516.	2612.	2
	1496.	2633.	2
	1491.	2514.	1

Hits were to be saved on targets #6, #7, #11, and #13; only #7 was hit

TARGET NUMBER 1	X-DIM	Y-DIM	WPN TYPE
	3060.	2201.	1
	3040.	2202.	1
	3270.	2182.	1
	3213.	2193.	1
	3275.	2209.	1
	3269.	2174.	1
	5523.	3601.	1
	5565.	3641.	1
	5709.	3720.	1
	5764.	3662.	1
	5697.	3697.	1
	5834.	3600.	1
	6013.	3701.	1

TARGET NUMBER 2	X-DIM	Y-DIM	WPN TYPE
	1964.	2503.	1
	2038.	2567.	1
	2071.	2551.	1
	2118.	2638.	1
	2304.	2865.	1
	2500.	3012.	1

Fig. 10 — Target hit summary, Case 2

```
*****AIDA*****     AN AIRBASE DAMAGE ASSESSMENT MODEL
' DEVELOPED BY THE RAMSTEIN OFFICE OF THE RAND CORPORATION
NO OF TRIALS   1 NPRINT  0  MODE -2  MCL      0( 250)   MCW     0( 5)    MIN REPAIR 0  PLOT HITS 0   TEST  0
```

Directs use of the Expected-Value Mode

***** BASE COMPLEX NAME - BASE XYZ *****

TARGET DATA

NUMBER	X-DIM	Y-DIM	NE LIMB	SE LIMB	ANGLE	TGT TYPE	STORE	BLDG NO
1	1000.	1000.	8000.	150.	60.	1.	0.	RUNWAY
2	775.	1350.	5000.	50.	45.	1.	0.	MAIN TWY
3	540.	1610.	1750.	375.	45.	3.	0.	RAMP A
4	800.	1325.	50.	375.	60.	2.	0.	TXWY #1
5	4400.	4800.	3300.	50.	78.	2.	0.	TXWY #2
6	400.	1750.	600.	250.	45.	4.	1.	AC MT B1
7	1100.	2450.	400.	200.	45.	4.	1.	AC MT B2
8	2000.	3000.	250.	150.	45.	5.	0.	AV MT B3
9	2450.	3220.	500.	60.	45.	4.	0.	OPS B4
10	2850.	3800.	400.	250.	45.	5.	0.	AGE B5
11	1425.	3075.	300.	175.	45.	4.	1.	ENG B6
12	1580.	3325.	150.	80.	45.	5.	0.	ELEC B7
13	1930.	3575.	225.	100.	45.	5.	1.	WPNS B8
14	$\boxed{1200.}$	$\boxed{2000.}$	0.	0.	$\boxed{1600.}$	$\boxed{21.}$	0.	1988.

Special target type #21 added: requests a hit-density grid over a 1600' x 1600' square with the southwest corner at x=1200, y=2000

ATTACK DATA

NUMBER	HDG	X-DMPI	Y-DMPI	WTP	DEP	P-DISP	D-DISP	NO WPNS	LENGTH	WPN TYPE	ARRIVAL
1	45.	1775.	3350.	400.	200.	50.	30.	25.	800.	1.	1.000
2	45.	650.	1850.	151.	117.	28.	23.	6.	86.	2.	1.000
3	45.	1350.	2575.	151.	117.	28.	23.	6.	86.	2.	1.000
4	90.	2750.	1900.	400.	200.	50.	30.	25.	1500.	1.	1.000
5	90.	2750.	1900.	400.	200.	50.	30.	25.	1500.	1.	1.000
6	90.	6250.	3875.	400.	200.	50.	30.	25.	1500.	1.	1.000
7	90.	6250.	3875.	400.	200.	50.	30.	25.	1500.	1.	1.000
8	45.	2600.	3400.	400.	200.	50.	30.	25.	1200.	1.	1.000
9	45.	2600.	3400.	400.	200.	50.	30.	25.	1200.	1.	1.000

Note that CBU attack was dropped since this mode will not function if a CBU is in attack

MISS DISTANCES ALLOWED FOR EFFECTIVE HITS

TARGET TYPES

WPN TYPE	WPN REL	1	2	3	4	5	6	7	8	9	10
1	0.950	22.	22.	22.	40.	40.	0.	0.	0.	0.	0.
		0.0	0.0	0.0	50.000	75.000	0.0	0.0	0.0	0.0	0.0
2	0.950	28.	28.	28.	50.	50.	0.	0.	0.	0.	0.
		0.0	0.0	0.0	60.000	90.000	0.0	0.0	0.0	0.0	0.0

Fig. 11 — Input data, Case 3

EXPECTED HIT DENSITY PER 10000 SQ FT

For different sized facilities, estimate the expected hits in proportion to their size

Y LOC	X - LOCATION 1200	1300	1400	1500	1600	1700	1800	1900	2000	2100	2200	2300	2400	2500	2600	2700	2800
3600	0.028	0.046	0.068	0.096	0.127	0.159	0.188	0.213	0.233	0.252	0.270	0.291	0.315	0.337	0.353	0.357	0.345
3500	0.042	0.065	0.093	0.126	0.160	0.192	0.221	0.245	0.265	0.284	0.305	0.328	0.351	0.369	0.377	0.368	0.342
3400	0.059	0.086	0.119	0.154	0.189	0.220	0.247	0.269	0.289	0.309	0.331	0.354	0.374	0.384	0.380	0.358	0.319
3300	0.076	0.107	0.142	0.178	0.211	0.239	0.263	0.283	0.303	0.324	0.346	0.366	0.378	0.378	0.361	0.327	0.279
3200	0.093	0.126	0.161	0.194	0.223	0.247	0.267	0.286	0.306	0.326	0.346	0.360	0.363	0.351	0.323	0.280	0.229
3100	0.108	0.142	0.175	0.205	0.228	0.246	0.262	0.279	0.298	0.317	0.331	0.337	0.330	0.308	0.272	0.226	0.177
3000	0.123	0.159	0.192	0.216	0.230	0.240	0.250	0.264	0.280	0.294	0.302	0.299	0.283	0.254	0.215	0.171	0.128
2900	0.144	0.187	0.221	0.237	0.237	0.232	0.233	0.241	0.253	0.261	0.262	0.251	0.229	0.197	0.160	0.122	0.088
2800	0.177	0.233	0.267	0.268	0.246	0.223	0.212	0.214	0.219	0.222	0.216	0.200	0.176	0.146	0.114	0.084	0.058
2700	0.220	0.286	0.314	0.294	0.248	0.208	0.187	0.183	0.184	0.181	0.171	0.154	0.131	0.106	0.081	0.059	0.041
2600	0.250	0.314	0.328	0.289	0.229	0.182	0.160	0.154	0.152	0.147	0.136	0.120	0.101	0.082	0.064	0.050	0.039
2500	0.243	0.293	0.291	0.236	0.190	0.152	0.136	0.131	0.130	0.125	0.116	0.104	0.091	0.078	0.068	0.059	0.053
2400	0.198	0.228	0.219	0.184	0.148	0.127	0.121	0.122	0.122	0.121	0.116	0.110	0.103	0.097	0.092	0.088	0.085
2300	0.140	0.155	0.149	0.133	0.121	0.117	0.121	0.127	0.133	0.136	0.138	0.138	0.138	0.138	0.138	0.137	0.135
2200	0.095	0.104	0.107	0.108	0.112	0.122	0.134	0.146	0.157	0.167	0.176	0.183	0.189	0.194	0.197	0.199	0.198
2100	0.072	0.080	0.090	0.102	0.117	0.134	0.152	0.170	0.187	0.204	0.219	0.232	0.243	0.252	0.259	0.262	0.261
2000	$\boxed{0.063}$	0.072	0.086	0.104	0.123	0.145	0.167	0.190	0.212	0.233	0.253	0.270	0.285	0.297	0.305	0.309	0.309

Southwest corner X = 1200, Y = 2000

CYCLES IN SPIDEN 16113

(Counts uses of EXP function)

Grid dimensions 1600′ × 1600′

Fig. 12 — Expected hit density, Case 3

hits per 10,000 sq ft, or, in effect, within the walls of a 100 × 100 ft building centered at the grid position.

Figure 13 presents the expected hits for the regular target array. As a comparison will quickly indicate, the particular results for the single trial in the second case are substantially different from the average result that is to be expected. Furthermore, as comparisons with the statistical averages presented in Case 1 will confirm, a rather large number of trials are required if one is to obtain reasonably accurate estimates of a mean in problems of the sort examined here using Monte Carlo techniques. It is for reasons such as this that it was suggested earlier that the two modes can be used together beneficially so as to more efficiently obtain both a reasonably good estimate of the mean as well as useful evidence on the variability about that mean.

Case 4 examines the availability of a minimum operating surface for aircraft. Since this type of problem is rather complex and tends to demand relatively large numbers of trials, it will often be preferable to separate this examination from others, as we have here in Cases 1 and 4, to avoid unnecessarily processing data relating to attacks and targets that are not of interest. For this reason all targets but the two of interest have been dropped (see Fig. 14), and all attacks that could not have any effect should also be dropped; for this case we retained all the attacks, since none are extremely distant from both surfaces.

The first result in Case 4 is the trial-by-trial record (top of Fig. 15) of the total number of hits and minimum numbers of repairs required for both of the targets; there is no entry only if there are no hits. This record is especially useful if one wishes to examine the distribution of attack results in more detail than can be done using the overall statistical results themselves. The latter are presented at the bottom of Fig. 15. As can be seen by comparing these results for the taxiway with those in the previous case, even 25 trials is inadequate to assure a reasonable estimate of the mean (which is actually nearly twice that which has been obtained in this sample of 25 trials).

```
CYCLES  17731
```

Counts uses of EXP function

```
TARGET      HITS
 NO       EXPECTED

** TARGET TYPE #  1 **

   1        15.454        RUNWAY
   2         5.507       MAIN TWY
            ------
            20.962(0.998)

** TARGET TYPE #  2 **

   4         0.007        TXWY #1
   5         0.006        TXWY #2
            ------
             0.014(0.007)

** TARGET TYPE #  3 **

   3         8.242        RAMP A
            ------
             8.242(1.000)

** TARGET TYPE #  4 **

   6         4.251       AC MT B1
   7         3.850       AC MT B2
   9         2.797       OPS   B4
  11         2.072       ENG   B6
            ------
            12.970(0.944)

** TARGET TYPE #  5 **

   8         2.236       AV MT B3
  10         3.623       AGE   B5
  12         0.836       ELEC  B7
  13         1.203       WPNS  B8
            ------
             7.898(0.783)
```

Expected fraction of the facilities of this type with at least one hit

Total number of hits expected on targets of this type

Fig. 13 — Expected target damage, Case 3

```
*****************************************************************************************************************
                    *****AIDA*****    AN AIRBASE DAMAGE ASSESSMENT MODEL
                    DEVELOPED BY THE RAMSTEIN OFFICE OF THE RAND CORPORATION
*****************************************************************************************************************
NO OF TRIALS  25 NPRINT  4  MODE  1  MCL   4000( 250)   MCW   50(10)   MIN REPAIR 1  PLOT HITS 0   TEST  0
*****************************************************************************************************************
```

Calls for compressed listing of runway status by trial

```
***** BASE COMPLEX NAME - BASE XYZ *****
```

TARGET DATA

NUMBER	X-DIM	Y-DIM	NE LIMB	SF LIMB	ANGLE	TGT TYPE	STOFE	BLDG NO
1	1000.	1000.	8000.	150.	60.	1.	0.	RUNWAY
2	775.	1350.	5000.	50.	45.	1.	0.	MAIN TWY

Note that all targets other than the potential aircraft operating surfaces have been dropped as directed on the CONTrol card

ATTACK DATA

NUMBER	HDG	X-DMPI	Y-DMPI	REP	DEP	R-DISP	D-DISP	NO WPNS	LENGTH	WPN TYPE	ARRIVAL
1	45.	1775.	3350.	400.	200.	50.	30.	25.	800.	1.	1.000
2	45.	650.	1850.	151.	117.	28.	23.	6.	86.	2.	1.000
3	45.	1350.	2575.	151.	117.	28.	23.	6.	86.	2.	1.000
4	90.	2750.	1900.	400.	200.	50.	30.	25.	1500.	1.	1.000
5	90.	2750.	1900.	400.	200.	50.	30.	25.	1500.	1.	1.000
6	90.	6250.	3875.	400.	200.	50.	30.	25.	1500.	1.	1.000
7	90.	6250.	3875.	400.	200.	50.	30.	25.	1500.	1.	1.000
8	45.	2600.	3400.	400.	200.	50.	30.	25.	1200.	1.	1.000
9	45.	2600.	3400.	400.	200.	50.	30.	25.	1200.	1.	1.000

All attacks were retained to account for all possible hits

MISS DISTANCES ALLOWED FOR EFFECTIVE HITS

WPN TYPE	WPN REL	TARGET TYPES 1	2	3	4	5	6	7	8	9	10
1	0.950	22.	22.	22.	40.	40.	0.	0.	0.	0.	0.
		0.0	0.0	0.0	50.000	75.000	0.0	0.0	0.0	0.0	0.0
2	0.950	28.	28.	28.	50.	50.	0.	0.	0.	0.	0.
		0.0	0.0	0.0	60.000	90.000	0.0	0.0	0.0	0.0	0.0

Note: These different radii for type #1 and #2 weapons will be taken into account in assessing runway status. (However, it is unlikely that #2 weapons will actually be involved in this example as aim is too accurate.)

Fig. 14 — Input data, Case 4

TRIAL	1	TGT	1	HITS	14	REPAIRS	0
TRIAL	1	TGT	2	HITS	3	REPAIRS	3
TRIAL	2	TGT	1	HITS	25	REPAIRS	3
TRIAL	2	TGT	2	HITS	12	REPAIRS	12
TRIAL	3	TGT	1	HITS	15	REPAIRS	0
TRIAL	3	TGT	2	HITS	4	REPAIRS	4
TRIAL	4	TGT	1	HITS	15	REPAIRS	0
TRIAL	5	TGT	1	HITS	12	REPAIRS	0
TRIAL	5	TGT	2	HITS	11	REPAIRS	11
TRIAL	6	TGT	1	HITS	18	REPAIRS	3
TRIAL	6	TGT	2	HITS	3	REPAIRS	3
TRIAL	7	TGT	1	HITS	27	REPAIRS	3
TRIAL	8	TGT	1	HITS	24	REPAIRS	2
TRIAL	9	TGT	1	HITS	7	REPAIRS	0
TRIAL	9	TGT	2	HITS	1	REPAIRS	1
TRIAL	10	TGT	1	HITS	19	REPAIRS	0
TRIAL	11	TGT	1	HITS	5	REPAIRS	0
TRIAL	11	TGT	2	HITS	18	REPAIRS	14
TRIAL	12	TGT	1	HITS	9	REPAIRS	0
TRIAL	12	TGT	2	HITS	3	REPAIRS	3
TRIAL	13	TGT	1	HITS	19	REPAIRS	1
TRIAL	13	TGT	2	HITS	1	REPAIRS	1
TRIAL	14	TGT	1	HITS	21	REPAIRS	3
TRIAL	14	TGT	2	HITS	2	REPAIRS	0
TRIAL	15	TGT	1	HITS	14	REPAIRS	1
TRIAL	15	TGT	2	HITS	16	REPAIRS	16
TRIAL	16	TGT	1	HITS	23	REPAIRS	1
TRIAL	17	TGT	1	HITS	17	REPAIRS	2
TRIAL	18	TGT	1	HITS	11	REPAIRS	0
TRIAL	18	TGT	2	HITS	1	REPAIRS	1
TRIAL	19	TGT	1	HITS	21	REPAIRS	2
TRIAL	20	TGT	1	HITS	19	REPAIRS	2
TRIAL	21	TGT	1	HITS	11	REPAIRS	0
TRIAL	22	TGT	1	HITS	14	REPAIRS	1
TRIAL	22	TGT	2	HITS	15	REPAIRS	15
TRIAL	23	TGT	1	HITS	26	REPAIRS	5
TRIAL	23	TGT	2	HITS	15	REPAIRS	15
TRIAL	24	TGT	1	HITS	27	REPAIRS	1
TRIAL	24	TGT	2	HITS	1	REPAIRS	0
TRIAL	25	TGT	1	HITS	5	REPAIRS	0
TRIAL	25	TGT	2	HITS	5	REPAIRS	6

The 'total number of hits' and the minimum number of craters requiring repair to obtain a 4000' x 50' operating surface' are listed for each type #1 target that receives at least one hit

This listing is useful for plotting the distribution of the trial results when the average values and standard deviation, given below, are insufficient.

TARGET DAMAGE STATISTICS FOR 25 TRIALS

TARGET NUMBER	PERCENT ATTACKS HIT	AVERAGE HITS PER ATTACK	STD. DEV. OF HITS	AVG. CBU COVERAGE	STD. DEV. COVERAGE	AVG. BOMB COVERAGE EMP	AVG. BOMB COVERAGE OTHER	CBU PK	BLDG NO.
TARGET TYPE # 1									
1	100.0	14.24	6.08	0.0	0.0	0.0	0.0	0.0	RUNWAY
2	64.0	4.44	6.05	0.0	0.0	0.0	0.0	0.0	MAIN TWY
		------ 20.68		------ 0.0					

DAMAGE STATISTICS BY TARGET TYPE

TARGET TYPE	AVERAGE PERCENT HIT	STANDARD DEVIATION	---- COVERAGE ---- EMP	OTHER	CBU
1	82.0	24.5	0.0	0.0	0.0

AT LEAST ONE MINIMUM RUNWAY SECTION WAS OPEN AFTER 76.0 PERCENT OF THE ATTACKS

WHEN ALL RUNWAYS WERE CLOSED, 2.3(1.6) HOLES REQUIRED REPAIR, ON THE AVERAGE, TO PROVIDE A MINIMUM RUNWAY

Standard deviation

Note that the repair requirement is not averaged over all trials but only those in which the minimum surface was not available.

Fig. 15 — Trial-by-trial results and target damage statistics, Case 4

Total computation time for these four cases was just under 53 sec on USAFE's WWMCCS Honeywell 6060 computer. If not overlaid, 36K words of core are required for AIDA as presently dimensioned, that is, for 250 targets, 50 attacks, 20 target types, and 10 weapon types.

Appendix A

DETAILED DESCRIPTION OF AIDA INPUT

The basic types of input cards employed with AIDA are as noted below:

CONT	control card
TGT	target card; one per target
ATT	attack card; one per weapon delivery pass (or group of identical passes)
ATT2	alternate attack card
EMD	effective miss distance card; one for each weapon type
REDO	controls sequential cases
END	terminates overall computation

The ATT2 card is actually two cards in sequence and the EMD card may have up to three supplementary cards. A detailed description of the entries for each type of card is presented on the pages that follow.

The general arrangement of data on all basic card types is similar; the card type-name is placed (left-adjusted) in the first four columns and the data are listed in eleven 6-column fields between Columns 7 and 72. All data are read with a F6.0 format; i.e., they are to be real numbers. If a whole number is to be input, it may be entered (right-adjusted) in the field without a decimal point; the decimal point is necessary otherwise. Columns 5 and 6 on the ATT, ATT2, and EMD cards are also used, as will be described, and the name of the target complex being studied and a name for each target may be included in Columns 73 through 80 of the CONT and TGT cards, respectively; any alphanumeric names are acceptable.

All linear dimensions should be in consistent units* (e.g., feet) and the target orientation and the attack heading entries should be in degrees.

---

*If ATT2 cards are to be used, all linear dimensions must be in feet.

CONT

The CONT card controls the mode of operation, the choice of random number generator, the number of trials (attack replications), and printout options; specifies the minimum clear length (MCL) and minimum clear width (MCW) for runway attack effectiveness calculations; and controls the runway repair assessment.

Columns	Data Entry
1-4	CONT
11-12	When 0, the seed for the random number generator is the same for all runs. If greater than 0, the seed is changed from run to run; if equal to -1, the random number generator is locked out. If equal to -2, the expected-value mode of operation replaces the Monte Carlo mode.
13-18	Desired number of replications. Default is 1.
23-24	Controls printout options as follows. If entry is 5 Prints multiple trial statistics plus a condensed listing of hits by trial 4 Prints multiple trial statistics plus a condensed listing of runway status by trial 3 Prints multiple trial statistics only 2 Above plus runway results for each trial 1 All above plus hit summary for each trial 0 All above plus stored hit data for each trial -1 All above plus all hits and target corners -2 All above plus all impact points
30	Controls printout of intermediate information for program test purposes; should normally be 0. If set to greater than 7, the random number generator is locked out. See the program source listing for the effect of other values.
31-36	MCL for aircraft operations. (Used to test if the runways are open.)
37-42	MCW for aircraft operations. (Used to test if the runways are open.)

Columns	Data Entry
48	When entry is 1, runway results will include the minimum number of craters to be repaired for the runway to meet the MCL and MCW criteria.
54	When the entry is 1, a plot of all impact points will be included for all closed runways (if, also, the printout option entry in Columns 23 and 24 is less than 3); when the entry is 2, impact plots are provided for each runway whether or not it is closed.
55-60	The distance that the "minimum runway rectangle" is to be shifted laterally in checking for an adequate section; the default value is 5.
61-66	The distance along the runway that the minimum runway rectangle is to be shifted in checking for an adequate section; the default value is 250.
73-80	A name can be entered here for the entire target complex and it will appear in the heading of the output listing.

TGT

Each TGT card designates the location, size, and orientation of a rectangular target.

Columns	Data Entry
1-3	TGT
7-12	The X-coordinate of the westernmost corner of the target.
13-18	The Y-coordinate of the westernmost corner of the target. If a target boundary runs exactly north-south, the X and Y coordinates of the southwestern corner should be specified.
19-24	Target dimension along the boundary running northeast (or north) from the X and Y coordinates of the reference corner specified in the two previous fields.
25-30	Target dimension along the boundary running southeast (or east) from the reference corner.
31-36	Heading in degrees of the northeast (or north) heading boundary of the target (along the dimension specified in Columns 19 to 24). (Meaning varies for target type #21; see below.)
41-42	Target type. Targets may be grouped into up to 10 (or 20) different categories with like vulnerabilities. This entry is used in conjunction with the effective miss distance on the EMD card. Target type #1 is restricted to runways and taxiways that may be used for flight operations; there will be no more than 5 targets of this type. Entering a 21 for target type actually acts as a signal (but only in conjunction with the expected-value mode) directing that a 17 × 17 grid of hit-density values be tabulated over a square, the southwest corner of which is entered in Columns 7 to 12 and 13 to 18. In this case, entries in the third, fourth, and seventh fields have no meaning. Unless a diffeıent value is entered in Columns 31 to 36 (preferably a number divisible by 16), the default dimension of the square is 4000, for a grid increment of 250.

Columns	Data Entry
	There may be one or more target type #21 cards, and they may be intermingled with normal target cards; however, when present, one of the type #21 cards *must* be the last target card entered for a case.
48	If greater than 0, all hit locations will be saved (and printed when entry in Column 24 of the CONT card is 0 or less).
73-80	A name or number for the target (any alphanumeric) may be entered here. This name as well as the sequence number that is assigned automatically will appear for target identification in the output listing.

ATT

The ATT card specifies the parameters of each weapon-delivery pass. Inputs required are the attack heading (measured from north in the coordinate system used to specify the targets), the desired mean point of impact (DMPI) for a single weapon or for the middle of a stick of weapons, the aiming error expressed as REP and DEP, the ballistic error of the individual weapons, the number of weapons to be delivered in the pass, the stick length, and the weapon type (related to the effective miss distance on the EMD card).

Columns	Data Entry
1-3	ATT
5-6	Total number of passes with the following characteristics; default = 1.
10-12	Attack heading in degrees from north.
13-18	The X-coordinate of the DMPI of a single weapon or the middle of a stick of weapons.
19-24	The Y-coordinate of the DMPI as above.
25-30	The REP.
31-36	The DEP.
37-42	Ballistic dispersion in range of individual weapons (R-DISP).
43-48	Ballistic dispersion in deflection of individual weapons (D-DISP). Default value is R-DISP.
49-54	The number of weapons in the stick.
55-60	The length of the stick (the distance between the first and last weapon of the stick in the absence of dispersion).
61-66	The weapon type (used in effectiveness calculations together with EMD and target type). An entry is required (an integer from 1 to 10); otherwise hits will not be recorded.
67-72	Probability of arrival at target; default = 1.0.

ATT2

The ATT2 card should be used in place of the ATT card when the user wishes assistance with trajectory calculations. When this card is used the user expresses the attack in terms of speed, altitude, dive angle, intervalometer settings, etc., and a special subroutine converts these inputs to those demanded on the ATT card. The conversion procedure is the JMEM/AS Open End Method Zero as outlined in the *Users Manual for JMEM/AS Open-End Methods*, WANG Labs., Inc., Tewksbury, Mass., August 1974.

Both ATT and ATT2 type cards may be used in the same run; the order of entry is of no importance. When ATT2 cards are used the input data will be reproduced as submitted, as well as being tabulated in the normal manner, after conversion.

Data input with the ATT2 procedure require *two* cards. The first card is labeled ATT2 in the first 4 columns and has input similar to that on an ATT card (all fields are read with a F6.0 format); a second unlabeled card is mandatory following each ATT2 card. The format for both cards follows. When these cards are used, all linear dimensions in the input data will be in feet.

Columns	Data Entry
1-4	ATT2
5-6	Total number of passes with the following characteristics; default = 1.
10-12	Attack heading in degrees from north.
13-18	The X-coordinate of the DMPI of a single weapon or the middle of a stick of weapons.
19-24	The Y-coordinate of the DMPI as above.
25-30	The CEP in the normal plane in *mils*, or, if DEP is specified, a constant which, when divided by the sin of the impact angle, gives the REP, in *mils*.
31-36	The DEP in *mils* (if omitted, CEP controls).
37-42	Ballistic dispersion in *mils*.
49-54	The number of weapons in the stick.
61-66	The weapon type.
67-72	Probability of arrival at target; default = 1.0.

The data format for the second card of each ATT2 pair is as noted below (this card is used with a 6F6.0, 3F6.3 format). Typical ballistic data required for this card are noted in Table A-1.

Columns	Data Entry
7-12	Aircraft velocity (kn).
13-18	Release altitude of last bomb (ft).
19-24	Dive angle at release (deg).
25-30	Terminal velocity of weapon (cluster) or first leg of a high-drag bomb (ft/sec) (VT1 in JMEM).*
31-36	Terminal velocity of a cluster bomblet or a high-drag bomb (ft/sec) (VT2 in JMEM).
37-42	Probable error in estimating and correcting for wind effects (ft/sec).
43-48	Cluster opening time or fin opening time for a high-drag bomb (*ms*), or cluster/fin opening altitude (ft). (A *decimal* point is mandatory when altitude is input.) (TD or H_f in JMEM.)
49-54	Intervalometer setting (*ms*).
55-60	Dispensor intervalometer setting (*ms*) (0 for clusters).

*Illustrative values are noted on Table A-1.

Table A-1

TYPICAL BALLISTIC PARAMETERS

Weapon	VT_1 (fps)	Ballistic Parameter	
		VT_2 (fps)	T_D or H_f
Mk-81 Mod 1	1850	0	0
Mk-81 SE	1100	208	300 ms
Mk-82 Mod 1	1900	0	0
Mk-82 SE	1200	240	350 ms
Mk-83	2500	0	0
Mk-84	2850	0	0
M-117 Unretarded	1950	0	0
M-117 Retarded	900	168	300 ms
M-118	2450	0	0
AN-M64A1	1600	0	0
AN-M65A1	2000	0	0
Mk-36 DST	1200	240	350 ms
CBU-38	450	0	0
CBU-52B/B	1000	230	Variable Altitude (ft)
CBU-58/B	950	215	Variable Altitude (ft)

SOURCE: *Users Manual for JMEM/AS Open-End Methods*, WANG Labs., Inc., Tewksbury, Mass., August 1974.

EMD

The EMD card is optional and provides information regarding weapon performance against the various types of targets. The entries for this card are different for point-impact weapons and for CBU-type munitions. For point-impact weapons, a hit is assessed for any impact within a distance of EMD from the target. For CBU munitions, the EMD card is used to specify the dimensions of the rectangular bomblet pattern.

The methods for expressing weapon coverage also differ for the two types of munitions. For point-impact weapons the EMD is also used as the weapon kill radius, and coverage is determined as that fraction of the target area that is covered by a circle of that radius.

For point-impact weapons (GP bombs or PGMs) the entries are:

Columns	Data Entry
1-3	EMD
5	Enter 1 if data are to be entered for 20 target types.*
6	Enter 1 if data on weapon reliability, p_k, or effective kill radius for this weapon type, are to be entered (on the following card).
11-12	Weapon type (used in conjunction with Columns 61 to 66 on ATT card).
13-18	EMD for point-impact weapons versus target type #1.
19-24	EMD versus target type #2.
67-72	EMD versus target type #10.

If the weapons are CBU-type munitions, use the following entries on the EMD card.

Columns	Data Entry
1-3	EMD
5	Enter 1 if data are to be entered for 20 target types.

*When more than 10 target types are involved, the EMD data and, if specified, the supplemental coverage data (see p. 48), for target types #11 through #20 are entered in ten 6-column fields from Columns 13 to 72 on cards that immediately follow the EMD card (and supplementary card).

Columns	Data Entry
6	Enter 1 if data on weapon reliability and/or on kill probabilities are to be entered for any target type on the following card.
11-12	Weapon type (used in conjunction with Columns 61 to 66 on ATT card).
13-18	Enter CBU pattern length as a *negative* entry.
19-24	Enter CBU pattern width as a *positive* entry.

SUPPLEMENTAL CARD FOR WEAPON RELIABILITY AND COVERAGE FACTORS

If a 1 is entered in Column 6 of an EMD card, a supplemental card must be included next with the weapon reliability and a set of entries for the several target types. Note that this card is not identified, but one must follow each EMD card that has an entry in Column 6. If a 1 is entered in Column 5 of an EMD card, as well as in Column 6, a second supplementary card is required for target types #11 through #20; this card is the fourth of four.

All entries on these cards are optional; the default value for reliability is 1.0. If an entry is made in any of the last 10 (20) fields and it is not greater than unity, it is taken as the user estimate of the p_k for that particular weapon-target combination for either point-impact weapons or CBU munitions. For point-impact weapons, an entry that exceeds unity is taken as an additional kill radius and another coverage fraction is determined as that fraction of the target area that is covered by a circle of that radius, given a hit within EMD of the target. Thus, when there are entries on the supplemental card for certain target types, coverage fractions are computed both for the corresponding value of EMD as well as for the value on the supplemental card.

Columns	Data Entry
7-12	Reliability* of this weapon type; default = 1.0.
13-18	p_k or kill radius† for this weapon versus target type #1.
19-24	p_k or kill radius for this weapon versus target type #2.
	.
	.
	.
67-72	p_k of this type weapon versus target type #10.

Entries for target types #11 through #20 on a second supplemental card will be in the ten 6-column fields between Column 13 and Column 72.

---

*Since these entries are read with an F6.0 format, the decimal point must be included.

†Only for point-impact weapons.

REDO

The REDO card is used to terminate the input for one case and initiate a new case with some or all of the previous inputs, as described earlier.

Columns	Data Entry
1-4	REDO
7-12	Number of prior targets to be retained. All will be retained if there is no entry. Use a negative entry if none are to be retained.
13-18	Number of prior attacks to be retained. All will be retained if there is no entry. Use a negative entry if none are to be retained.
19-24	An entry of unity suppresses the input listings for targets and/or for attacks and weapons if no changes have been made in these data sets from the prior case.

END

An END card must be included at the end of all data entry cards.

Columns	Data Entry
1-3	END

Appendix B

GLOSSARY OF TERMS USED IN AIDA

KEY VARIABLES

INL — Distance along the runway the "minimum runway rectangle" is shifted.

INW — Lateral distance the minimum runway rectangle is shifted in checking for an adequate section.

ITRIAL — Number of the current trial.

KCBU — Switch; set to unity if any weapons are CBUs.

KPTI — Switch; set to unity if any weapons are the point-impact type.

KTEST — Index controlling variety of debugging printout options.

LIST — Switch; when set to unity, target and/or attack input lists are suppressed when unchanged.

MCL — Minimum adequate length for required runway.

MCR — Switch; set to unity when runway availability is to be checked.

MCW — Minimum adequate width for required runway.

MODE — Index controlling mode of operation (see pp. 9 and 38).

MTT — Largest target type number in the target array.

NA — Total number of weapon-delivery passes.

NAM — Maximum permissible number of weapon-delivery passes.

ND — Number of types of weapons in overall attack.

NHITD — Switch; set to unity when the expected-value mode is specified.

NJMEM — Number of weapon-delivery passes that required trajectory calculations.

NPLOT	Switch; set to 1 or 2 if runway impact plots are desired.
NPRINT	Index controlling results output (see pp. 9 and 38).
NREDO	Switch; set to unity if an additional case is specified.
NREP	Switch; set to unity when repair requirements are to be assessed.
NSAVE1	Number of targets to be retained for a subsequent case.
NSAVE2	Number of weapon-delivery passes to be retained for a subsequent case.
NST	Maximum number of targets for which hits can be stored.
NSTAT	Cumulative number of trials in which the minimum runway was available.
NT	Total number of targets.
NTM	Maximum permissible number of targets.
NTRIAL	Total number of trials specified.

KEY ARRAYS

AMD(I,J,K)	Weapon effectiveness data.
I	Weapon type.
J	Target type.
K = 1	Effective miss distance.
2	Effective damage radius or probability of kill.
ATT(I,J)	Strong array for weapon-delivery data.
I	Weapon-delivery pass number; numbered internally in order of entry.
J = 1	Heading (deg).
2	X coordinate of desired mean point of impact.
3	Y coordinate of DMPI.
4	Range error probable of DMPI.
5	Deflection error probable of DMPI.
6	Dispersion in range (ground plane).
7	Number of weapons released in pass.
8	Length of stick (in ground plane).
9	Weapon type.
10	Dispersion in deflection.
11	Probability attacker arrives at target.
CBUHT(J,K)	Impact coordinates of the centroid of the J*th* CBU pattern.
K = 1	X coordinate.
2	Y coordinate.
COV(L)	Fraction of target L covered by one or more CBU patterns.

HIT(I,J,K)	Storage array for hit locations on specified targets.
I	I*th* of those targets for which hit data are to be stored.
J = 1	X coordinate.
2	Y coordinate.
3	Weapon type.
K	Number of hit on the I*th* target.
HITR(I,J,K)	Storage array for hit locations on type #1 targets (i.e., runways and taxiways).
I,J,K	See HIT(I,J,K).
IR(N)	Switch; set to unity if the N*th* weapon-delivery attacker fails to reach target.
IZONE(K,J)	Denotes which of the ordered targets (see TO) fall in the K*th* target zone.
J = 1	Lowest numbered target in the K*th* zone.
2	Highest numbered target in the K*th* zone.
MHIT(K)	Target number of the K*th* target for which hit location data are to be stored.
MSTAT(J)	Storage array for accumulating trial results of runway availability tests.
J = 1	Minimum number of repairs required to open a minimum runway.
2	Square of J = 1, above.
3-8	Not used.
MTYPE(I)	Index that specifies whether or not supplementary data are to follow the EMD card for weapon type I.
NCBU(L)	Number of CBU weapon patterns that cover all or part of target L.

NHIT(L)	Number of hits on target L; by both point-impact and CBU weapons.
NRW(I)	Target number of the I*th* runway entered.
P(L,K)	
K = 1	Expected fraction of target L that is covered by the effects of point-impact weapons; or probability of kill of target L due to point-impact weapons.
2	Probability of kill of target L due to CBU weapon patterns.
STAT(L,J)	Storage array for accumulating trial results.
L	Target number.
J = 1	Number of hits by point-impact weapons.
2	Square of J = 1, above.
3	Trials with at least one hit.
4	Fractional coverage by CBU weapons.
5	Square of J = 4, above.
6	Expected fractional coverage by point-impact weapon; or probability of kill.
7	Probability of kill by CBU-type weapons.
8	Unused.
STAT2(I,J)	Storage array for accumulating trial results for targets of a given type.
I	Target type.
J = 1	Fraction of the targets of type I that received at least one hit.
2	Square of J = 1, above.
TGT(L,J)	Storage array for target data.

L	Target number; numbered internally in order of entry.
J = 1	X coordinate of westernmost corner (#1).
2	Y coordinate of corner #1.
3	X coordinate of corner #2.
4	Y coordinate of corner #2.
5	X coordinate of corner #3.
6	Y coordinate of corner #3.
7	X coordinate of corner #4.
8	Y coordinate of corner #4.
9	Heading of northeast target leg.
10	Target type.
11	Switch; hits stored when reset to unity.
12	Dimension of NE target leg.
13	Dimension of SE target leg.
TO(I,J)	Target order array in which targets are ordered according to increasing values of the sum of the coordinates of the western corner.
I	I*th* target in the ordered array.
J = 1	Value of (X+Y) for the I*th* ordered target.
2	Number of the target as initially entered.
WPNREL(I)	Reliability of weapon type I.

Appendix C

PROGRAM LISTING

```
C     MAIN-AIDA AIRBASE DAMAGE ASSESSMENT MODEL
         COMMON /ARRAYS/ TGT(250,13), ATT(50,11),AMD(10,20,2), TO(250,2),
     X IZONE(50,2),NHIT(250),MHIT(20),HIT(20,3,25),NRW(5),HITP(5,3,250)
     X , P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNREL(10),NCBU(250)
         COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,MCW,MCL,MODE,NPRINT,NAM,NST,MTT
         COMMON/STATS/NTRIAL,ITRIAL,NSTAT,STAT(250,8),STAT2(20,5),MSTAT(8)
         COMMON /CONTRL/ NREP,NPLOT,INW,INL,NSAVE1,NSAVE2,LIST,NUMEM
         COMMON / HITON / NHITO, NREDO
         COMMON /CBUHIT/ CBUHIT(200,2), IR(50), KCBU, KPTI
         NT = 0
C     *****  NTM IS THE MAXIMUM NUMBER OF TARGETS  *****
C             ARRAYS:  TGT, IO, NHIT, P, COV, NAME, NCBU, STAT
         NTM = 250
C     *****  NAM IS THE MAXIMUM NUMBER OF ATTACKS
C             ARRAYS:   ATT, IR
         NAM = 50
C     *****  NST IS THE NUMBER OF TARGETS FOR WHICH HITS CAN BE STORED.
C             ARRAYS:  HIT, MHIT
         NST = 20
C     *****  TO CHANGE ANY OF THE PRECEDING DIMENSIONS, MAKE THE APPROPRIATE
C     *****  CHANGES IN THE ARRAYS AND THEN CHANGE THE LIMITING VALUE.
         NA = 0
         ND = 0
         MCR = 0
         NTRIAL = 1
         KTEST = 0
         NREDO = 0
         DO 5   I = 1,10
         MTYPE(I) = 0
         WPNREL(I) = 1.0
         DO 5   J = 1,20
         DO 5   K = 1,2
    5    AMD(I,J,K) = 0.0
   10    ITRIAL = 0
         DO 12   I = 1, 5
   12    NRW(I) = 0
         CALL INPUT
         IF (NHITO .EQ. 1)   GO TO 40
         IF (NTRIAL .LT. 2)   GO TO 25
         NSTAT = 0
         DO 15   I = 1,NT
         DO 15   N = 1,8
   15    STAT(I,N) = 0.0
         DO 20   M = 1,20
         DO 20   N = 1,5
   20    STAT2(M,N) = 0.0
   25    DO 30   I = 1, NST
   30    MHIT(I) = 0
   40    CONTINUE
         DO 45   N = 1,8
   45    MSTAT(N) = 0
         CALL TGTDIM
         IF (NHITO .EQ. 0)   GO TO 60
         IF (TGT(NT,13) .EQ. 21.)    CALL GRIDEN
         DO 50   I = 1,NT
         COV(I) = 0.0
         P(I,1) = 0.0
         P(I,2) = 0.0
   50    P(I,3) = 0.0
         CALL EXPHIT
```

```
      GO TO 200
   60 IF (KPTI .EQ. 0)   GO TO 100
      CALL TGTCRD
      CALL TGTCM
  100 CONTINUE
      ITRIAL = ITRIAL + 1
      DO 105  I1 = 1, NST
      DO 105  I2 = 1,3
      DO 105  I3 = 1, 25
  105 HIT(I1,I2,I3) = 0.0
      DO 110 I=1, NT
      COV(I) = 0.0
      P(I,1) = 0.0
      P(I,2) = 0.0
      P(I,3) = 0.0
      NCBU(I) = 0
  110 NHIT(I)=0
      DO 115  I1 = 1, 5
      DO 115  I2 = 1, 3
      DO 115  I3 = 1, 250
  115 HITS(I1,I2,I3) = 0.0
      CALL BOMB
      IF (KCBU .EQ. 1)   CALL CBU
      IF (MCS .EQ. 0) GO TO 130
      CALL CHECKS
  130 CONTINUE
      IF (NTRIAL .LT. 2)   GO TO 170
      DO 140   I = 1, NT
      AID = NHIT(I) - NCBU(I)
      STAT(I,1) = STAT(I,1) + AID
      STAT(I,2) = STAT(I,2) + AID*AID
      IF ( AID .GT. 0.0)   STAT(I,3) = STAT(I,3) + 1.
      STAT(I,4) = STAT(I,4) + COV(I)
      STAT(I,5) = STAT(I,5) + COV(I)*COV(I)
      STAT(I,6) = STAT(I,6) + P(I,1)
      STAT(I,7) = STAT(I,7) + P(I,2)
      STAT(I,8) = STAT(I,8) + P(I,3)
  140 CONTINUE
      DO 160   M = 1,20
      MM = 0
      NN = 0
      AID1 = 0.0
      AID3 = 0.0
      AID4 = 0.0
      AID5 = 0.0
      DO 150   I = 1,NT
      IF (TGT(I,10) .NE. M)   GO TO 150
      NN = NN + 1
      NAID = NHIT(I) - NCBU(I)
      AREA = TGT(I,12)*TGT(I,13)
      AID1 = AID1 + AREA
      AID3 = AID3 + AREA*P(I,1)
      AID4 = AID4 + AREA*P(I,2)
      AID5 = AID5 + AREA*P(I,3)
      IF (NAID .GT. 0)   MM = MM + 1
  150 CONTINUE
      IF (MM .EQ. 0)   GO TO 160
      AID = NN
      STA = MM/AID
      STAT2(M,1) = STAT2(M,1) + STA
      STAT2(M,2) = STAT2(M,2) + STA*STA
```

```
            STAT2(M,3) = STAT2(M,3) + AID3/AID1
            STAT2(M,4) = STAT2(M,4) + AID4/AID1
            STAT2(M,5) = STAT2(M,5) + AID5/AID1
  160       CONTINUE
  170       CONTINUE
            IF (NPRINT .GT. 1)   GO TO 180
            CALL PRINT
  180       CONTINUE
            IF (NPRINT .LT. 5)   GO TO 190
            DO 185   L = 1,NT
  185       WRITE(6,1001)  ITRIAL, L, NHIT(L)
  190       CONTINUE
            IF (ITRIAL .LT. NTRIAL)   GO TO 100
            IF (NTRIAL .GT. 1)  CALL STATIS
  200       IF (NREDO .EQ. 1)   GO TO 10
            STOP
 1001       FORMAT(' ','TRIAL',I4,'   TGT',I4,'    HITS',I4)
            END
            SUBROUTINE INPUT
            INTEGER*4 LABEL, AN, NAME1, NAME2, NAME, NBASE1, NBASE2
            COMMON /ARRAYS/ TGT(250,13), ATT(50,11),AMD(10,20,2), TD(250,2),
           XIZONE(50,2),NHIT(250),MHIT(20),HIT(20,3,25),NRW(5),HITR(5,3,250)
           X ,P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNREL(10),NCBU(250)
            COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,MCW,MCL,MODE,NPRINT,NAM,NST,MTT
            COMMON/STATS/NTRIAL,ITRIAL,NSTAT,STAT(250,8),STAT2(20,5),MSTAT(8)
            COMMON /CONTRL/ NREP,NPLOT,INW,INL,NSAVE1,NSAVE2,LIST,NJMEM
            COMMON /CBUHIT/ CBUHIT(200,2), IR(50), KCBU, KPTI
             COMMON / HITON / NHITO, NREDO
            DIMENSION LABEL(6), DATA(11)
            DATA LABEL /'TGT ','ATT ','ATT2','AMD ','CONT','REDO'/
            NJMEM = 0
            LIST1 = 0
            LIST2 = 0
            IF (NREDO .EQ. 0)    NHITO = 0
            IF (NREDO .EQ. 0)    GO TO 2
C          SEE NOTE AT LABEL '45'
            IF (NSAVE1 .GT. 0)    NT = NSAVE1
            IF (NSAVE2 .GT. 0)    NA = NSAVE2
            IF (NSAVE1 .LT. 0)    NT = 0
            IF (NSAVE2 .LT. 0)    NA = 0
            IF (NSAVE1 .EQ. 0)    LIST1 = LIST
            IF (NSAVE2 .EQ. 0)    LIST2 = LIST
            NSAVE1 = 0
            NSAVE2 = 0
    2       CONTINUE
            NREDO = 0
    6       READ (5,101) AN, NTYPE, (DATA(I), I=1,11), NAME1,NAME2
C          NO ENTRY IS REQUIRED IN COLUMNS 5 AND 6.  IF AN INTEGER IS FOUND
C          ON AN ATTACK CARD, THE ATTACK WILL BE REPEATED SO THAT THERE WILL
C          BE THAT TOTAL NUMBER OF ATTACKS WITH THE STATED CHARACTERISTICS.
C              (ONE ATTACK IS ASSUMED IF THERE IS NO ENTRY.)
C          IF AN ENTRY IS NOTED ON AN AMD CARD, RELIABILITY AND/OR PROBABILITY
C          OF KILL DATA WILL BE EXPECTED ON THE NEXT CARD.
            IF (AN .EQ. LABEL(1)) GO TO 10
            IF (AN .EQ. LABEL(2)) GO TO 20
            IF (AN .EQ. LABEL(3)) GO TO 26
            IF (AN .EQ. LABEL(4)) GO TO 30
            IF (AN .EQ. LABEL(5)) GO TO 40
            IF (AN .EQ. LABEL(6)) GO TO 45
            GO TO 50
   10       NT = NT + 1
```

```
      LIST1 = 0
      IF (NT .GT. NTM)   GO TO 120
      DO 12 I = 1,2
   12 TGT (NT,I) =DATA (I)
      DO 13 I = 3,4
   13 TGT(NT,I+9) = DATA(I)
      DO 14 I = 5,7
   14 TGT(NT,I + 4) = DATA (I)
      NAME(NT,1) = NAME1
      NAME(NT,2) = NAME2
      GO TO 6
   20 NA = NA + 1
      LIST2 = 0
      IF ( NA .GT. NAM)   GO TO 130
      DO 22   I = 1,6
   22 ATT(NA,I) = DATA(I)
      ATT(NA,10) = DATA(7)
      IF (DATA(7) .EQ. 0.0)   ATT(NA,10) = ATT(NA,6)
      IF (DATA(11) .EQ. 0.)   DATA(11) = 1.0
      ATT(NA,11) = DATA(11)
      DO 24   I = 7,9
   24 ATT(NA,I) = DATA(I+1)
      NTYPE = MAX0((NTYPE-1),0)
      IF (NTYPE .EQ. 0)   GO TO 6
      GO TO 20
   26 NA = NA + 1
      LIST2 = 0
      IF ( NA .GT. NAM)   GO TO 130
      IF (DATA(11) .EQ. 0.0)    DATA(11) = 1.0
      CALL JMEMO(NJMEM,DATA)
      DO 27   I = 1,11
   27 ATT(NA,I) = DATA(I)
   28 NTYPE = MAX0((NTYPE-1),0)
      IF (NTYPE .EQ. 0)   GO TO 6
      NA = NA + 1
      DO 29   I = 1,11
   29 ATT(NA,I) = ATT((NA-1),I)
      GO TO 28
   30 ND = ND + 1
      LIST2 = 0
      M = DATA(1)
      IF (M .GT. 10)   GO TO 140
      NTYPE(M) = NTYPE
      DO 31   N = 2,11
   31 AMD(M,N-1,1) = DATA(N)
      IF ((NTYPE .EQ. 0) .OR. (NTYPE .EQ. 10))   GO TO 34
      READ(5,114)  WR,(DATA(I), I=1,10)
      WPNREL(M) = WR
      IF (WR .EQ. 0.0)   WPNREL(M) = 1.0
      DO 32   N = 1,10
   32 AMD(M,N,2) = DATA(N)
   34 CONTINUE
C  THE FOLLOWING 8 STATEMENTS PERMIT 20 TARGET TYPES WHEN NEEDED
      IF (NTYPE .LT. 10)   GO TO 6
      READ (5,114)  (DATA(I), I = 1,11)
      DO 36   N = 11,20
   36 AMD(M,N,1) = DATA(N-9)
      IF (NTYPE .LT. 11)   GO TO 6
      READ(5,114)  (DATA(I), I = 1,11)
      DO 38   N = 11,20
   38 AMD(M,N,2) = DATA(N-9)
```

```
          GO TO 6
   40     MODE = DATA(1)
          IF (MODE.EQ.0) CALL RSTART(7)
          NTRIAL = DATA(2)
          NPRINT = DATA(3)
          KTEST = DATA(4)
          IF (NTRIAL .LT. 2)   NTRIAL = 1
          MCR = 0
          NHITD = 0
          IF (MODE .EQ. -2)    NHITD = 1
          IF (NHITD .EQ. 1)    DATA(5) = 0.0
          IF (DATA(5) .GT. 0.0)    MCR = 1
          MCL = DATA(5)
          MCW = DATA(6)
          NREP = DATA(7)
          NPLOT= DATA(8)
         INW = DATA(9)
         INL = DATA(10)
          IF (INW .EQ. 0)    INW = 5
         IF (INL .EQ. 0)    INL = 250
          NBASE1 = NAME1
          NBASE2 = NAME2
         GO TO 6
   45     NREDO = 1
C       THE FIRST TWO DATA ENTRIES ON THE REDO CARD MAY BE USED TO SPECIFY
C       THE NUMBERS OF PRIOR TARGETS AND ATTACKS, RESPECTIVELY, TO BE
C        INCLUDED IN THE NEW CALCULATION.  THOSE NUMBERS WILL BE SELECTED
C        IN RANK ORDER FROM THOSE INPUT PREVIOUSLY; IN NO CASE MAY THE
C        NUMBER BE LARGER THAN THE NUMBER AVAILABLE.   IF NO NUMBER IS
C        ENTERED, ALL PRIOR TARGETS AND ATTACKS WILL BE INCLUDED.  IF A
C        NEGATIVE NUMBER IS ENTERED (EG  -1), NONE OF THE PRIOR MEMBERS
C        OF THAT SET WILL BE TREATED.
          NSAVE1 = DATA(1)
          NSAVE2 = DATA(2)
C        IF THE THIRD ENTRY (COL 24)  IS SET TO UNITY, THE TARGET LIST
C        AND/OR THE ATTACK/WEAPON LISTS WILL BE SUPPRESSED IF
C        THEY HAVE NOT BEEN CHANGED.
          LIST = DATA(3)
   50     CONTINUE
          WRITE (6,111) NTRIAL, NPRINT, MODE, MCL, INL, MCW, INW,
        X   NREP, NPLOT, KTEST
          IF ((NBASE1 .EQ. 0) .AND. (NBASE2 .EQ. 0))   GO TO 55
          WRITE (6,100)    NBASE1, NBASE2
   55     CONTINUE
          IF (LIST1 .EQ. 1)    GO TO 65
          WRITE (6,102)
          MTT = 0
          DO 60 I = 1,NT
          IF ((TGT(I,10) .GT. MTT) .AND. (TGT(I,10) .NE. 21.))
        X     MTT = TGT(I,10)
   60     WRITE (6,112) I,(TGT(I,J),J=1,2),(TGT(I,J),J=12,13),
        X   (TGT(I,J),J=9,11),NAME(I,1),NAME(I,2)
   65     IF (LIST2 .EQ. 1)    GO TO 95
          WRITE (6,104)
          MWPN = 0
          KPTI = 0
          DO 70 I = 1, NA
          IF (ATT(I,9) .GT. MWPN)    MWPN = ATT(I,9)
          IF (AMD(ATT(I,9),1,1) .GE. 0)    KPTI = 1
   70     WRITE (6,103) I, (ATT(I,J),J=1,6),ATT(I,10),(ATT(I,J),J=7,9)
        X   , ATT(I,11)
```

```
          IF (ND .EQ. 0)   GO TO 95
          WRITE (6,105) (I, I=1,10)
          DO 90   I = 1, MWPN
          WRITE(6,106) I, WPNREL(I),(AMD(I,J,1),J=1,10)
          IF ((MTYPE(I) .EQ. 0) .OR. (MTYPE(I) .EQ. 10))   GO TO 80
          WRITE (6,107)  (AMD(I,J,2), J = 1,10)
   80     CONTINUE
          IF (MTYPE(I) .LT. 10)   GO TO 90
          WRITE(6,115)  (AMD(I,J,1), J=11,20)
          IF (MTYPE(I) .EQ. 10)   GO TO 90
          WRITE(6,107)   (AMD(I,J,2), J=11,20)
   90     CONTINUE
   95     CONTINUE
C        TARGET TYPE #1 IS RESERVED FOR RUNWAYS AND TAXIWAYS (OR OTHER
C        LARGE TARGETS IF MCP=0) AND HIT STORAGE IS PROVIDED FOR 250 HITS
C        BUT FOR A MAXIMUM OF FIVE TARGETS OF TYPE #1.
          NTX = 0
          DO 99   I = 1, NT
          IF (TGT(I,10) .NE. 1.)   GO TO 99
          NTX = NTX + 1
          IF (NTX .GT. 5)   GO TO 150
          NRW(NTX) = I
   99     CONTINUE
          RETURN
  120      WRITE (6,108)
           STOP
  130      WRITE (6,109)
           STOP
  140      WRITE (6,110)
           STOP
  150     WRITE (6,116)
          STOP
  100     FORMAT(' ',15X,'*****  BASE COMPLEX NAME - ',2A4,'  *****',/,/)
  101     FORMAT( A4,I2,11F6.0,2A4)
  102     FORMAT (' ', 20X,'TARGET DATA',/,' NUMBER        X-DIM     Y-DIM
        X NE LIMB   SE LIMB     ANGLE   TGT TYPE      STORE    BLDG NO',/,/)
  103     FORMAT(' ',I4,2X,10F10.0,F10.3)
  104     FORMAT(' ',/,  20X,'ATTACK DATA',/,' NUMBER        HDG     X-DMPI
        X  Y-DMPI        REP        CEP     R-DISP     D-DISP    NO WPNS     LENG
        XTH  WPN TYPE     ARRIVAL',//)
  105      FORMAT('1', 35X,'MISS DISTANCES ALLOWED FOR EFFECTIVE HITS',/,
        X50X,'TARGET TYPES',/,20X,10(6X,I2,2X),/,' WPN TYPE   WPN REL ',
        X/,/,/)
  106     FORMAT(' ',I6,7X,F5.3,10F10.0)
  107     FORMAT(' ', 21X,10F10.3)
  108     FORMAT(' ',/,'*****  TOO MANY TARGETS HAVE BEEN SPECIFIED  ****')
  109     FORMAT(' ',/,'*****  TOO MANY ATTACKS HAVE BEEN SPECIFIED ****')
  110     FORMAT(' ',/,'*****  TOO MANY TYPES OF WEAPONS HAVE BEEN ',
        X  'SPECIFIED  *****')
  111     FORMAT ('1',  130('*'),/,
        X  40X,'*****AIDA*****      AN AIRBASE DAMAGE ASSESSMENT MODEL',/,
        X  40X,'DEVELOPED BY THE RAMSTEIN OFFICE OF THE RAND CORPORATION',/
        X  ,130('*'),/,' NO OF TRIALS ', I3,' NPRINT ',I2,'  MODE ',I2,
        X  '  MCL ',I6,'(',I4,')  MCW ',I4,'(', I2,')   MIN REPAIR ',
        X I1,'  PLOT HITS ',I1,'   TEST ',I2,/,130('*'),/,/)
  112     FORMAT (' ', I4,4X,7F10.0,2X,2A4)
  114     FORMAT ( 6X,11F6.0)
  115     FORMAT (' ',18X, 10F10.0)
  116     FORMAT (' ',/,'*****  TOO MANY RUNWAYS/TAXIWAYS HAVE BEEN',
        X  ' SPECIFIED  *****')
          END
```

```
      SUBROUTINE TGTDIM
      COMMON /ARRAYS/ TGT(250,13), ATT(50,11),AMD(10,20,2), TO(250,2),
     XIZONE(50,2),NHIT(250),MHIT(20),HIT(20,3,25),NRW(5),HITR(5,3,250)
     X ,P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNREL(10),NCBU(250)
     COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,MCW,MCL,MODL,NPRINT,NAM,NST,MTT
      COMMON/STATS/NTRIAL,ITRIAL,NSTAT,STAT(250,8),STAT2(20,5),MSTAT(8)
      COMMON / HITDN / NHITD, NREDO
      IF (NPRINT .LT. 0)   WRITE (6,104)
      DO 20 I = 1, NT
C     TGTDIM COMPUTES AND STORES THE LOCATION OF THE OTHER CORNERS.
      L1 = TGT(I,12)
      L2 = TGT(I,13)
      THETA = TGT (I, 9)/57.3
      S = SIN (THETA)
      C = COS (THETA)
      L1S = L1*S
      L1C = L1*C
      L2S = L2*S
      L2C = L2*C
      TGT (I,3) = TGT (I, 1) + L1S
      TGT (I,4) = TGT (I, 2) + L1C
      TGT (I,5) = TGT (I, 3) + L2C
      TGT (I,6) = TGT (I, 4) - L2S
      TGT (I,7) = TGT (I, 5) - L1S
      TGT (I,8) = TGT (I, 6) - L1C
      IF ((KTEST .GT. 2) .OR. (NPRINT .LT. 0))
     X   WRITE (6,102) I,(TGT(I,K), K=1,8)
   20 CONTINUE
      IF (NHITD .EQ. 1)   GO TO 50
      NR = 0
      DO 30 I = 1, NT
C     FOR SPECIFIED TARGETS THE TARGET NUMBER IS STORED IN MHIT FOR
C     LATER REFERENCE.
      IF ((TGT(I,11) .LT. 1.) .OR. (TGT(I,10) .EQ. 1.))   GO TO 30
      NR = NR + 1
      IF (NR .GT. NST)  GO TO 80
      MHIT(NR) = I
      IF ((KTEST .GT. 4) .AND. (ITRIAL .LT. 2))
     X     WRITE (6,101)  NR, MHIT(NR)
   30 CONTINUE
      IF (NR .EQ. NST)   GO TO 50
      NR1 = NR+1
      DO 40   I = NR1, NST
   40 MHIT(I) = 0
   50 CONTINUE
      IF ((NPRINT .LT. 0) .OR. (KTEST .GT. 2))  WRITE(6,104)
      RETURN
   80 WRITE (6,103)
      STOP
  101 FORMAT( ' ','MHIT(' , I2, ') = ',I2)
  102 FORMAT(' ','TARGET CORNER : TGT # ',I4,4(4X,F6.0,1X,F6.0))
  103 FORMAT('0', '  COMPUTATION STOPPED:  HIT DATA SPACE ',
     X  'REQUIRED FOR MORE THAN "NST" TARGETS')
  104 FORMAT ('1')
         END
      SUBROUTINE TGTORD
      COMMON /ARRAYS/ TGT(250,13), ATT(50,11),AMD(10,20,2), TO(250,2),
     XIZONE(50,2),NHIT(250),MHIT(20),HIT(20,3,25),NRW(5),HITR(5,3,250)
     X ,P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNREL(10),NCBU(250)
      COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,MCW,MCL,MODE,NPRINT,NAM,NST,MTT
C     THIS ROUTINE CREATES AN ARRAY IN WHICH THE TARGET NUMBERS ARE
```

```
C     ORDERED ACCORDING TO INCREASING VALUES OF (X+Y) OF THE INDEX CORNER
C     OUTPUT VALUES : TO(I,1) IS THE (X+Y) AND TO(I,2) IS THE 'ORIGINAL'
C     TARGET NUMBER;  I.E.  ITS POSITION IN THE INPUT LIST
        DO 10  I=1,NT
C     INITIALIZES TO  (TARGET ORDER) ARRAY
        IF (TGT (I,9) .GT. 45.)  GO TO 5
        TO(I,1)=TGT(I,1)+TGT(I,2)
        GO TO 8
    5   TO(I,1)=TGT(I,7)+TGT(I,8)
    8   TO(I,2)=I
   10   CONTINUE
        DO 20 J=2,NT
        NTEST=0
        DO 15 K=2,NT
C     REORGANIZES THE TO ARRAY INTO INCREASING VALUES OF THE INDEX CORNER
        I=NT-K+2
        IF (TO(I,1)  .GE. TO(I-1,1))  GO TO 15
        NTEST=1
        T=TO(I-1,1)
        TN=TO(I-1,2)
        TO(I-1,1)=TO(I,1)
        TO(I-1,2)=TO(I,2)
        TO(I,1)=T
        TO(I,2)=TN
   15   CONTINUE
        IF (NTEST .EQ. 0)   GO TO 25
   20   CONTINUE
   25   CONTINUE
        IF (KTEST .LT. 3)   GO TO 40
        DO 30   I = 1, NT
        NTO = TO(I,2)
   30   IF (KTEST .GT. 5)   WRITE (6,101)   I , NTO
   40   CONTINUE
        NT1 = NT + 1
        DO 50   I = NT1, NTM
        DO 50   J = 1,2
   50   TO(I,J) = 0.0
        RETURN
  101   FORMAT (' ',' RANK  ',I3,'   TARGET # ', I3)
        END
        SUBROUTINE TGTZON
        COMMON /ARRAYS/ TGT(250,13), ATT(50,11),AMD(10,20,2), TO(250,2),
     XIZONE(50,2),NHIT(250),MHIT(20),HIT(20,3,25),NRW(5),HITR(5,3,250)
     X ,P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNREL(10),NCBU(250)
      COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,MCW,MCL,MODE,NPRINT,NAM,NST,MTT
        DIMENSION IZ(250)
C     TGTZON IDENTIFIES TARGET LOCATION IN TERMS OF ITS IZONE SO THAT THE
C     SUBSEQUENT SEARCH PROCESS CAN BE REDUCED.   CONSIDER THE ENTIRE
C     TARGET AREA MAPPED BY LINES OF CONSTANT (X+Y).  ALL TARGETS WITH '
C     'INDEX CORNER'  (X+Y)  FALLING INTO THE K TH 500 FOOT SEGMENT OF
C     (X+Y) ARE IN THE K TH ZONE.   THE ORDERED INDEX NUMBER FOR THE
C     TARGET WITH THE LOWEST (X+Y) IN THE ZONE IS IZONE(K,1); THAT WITH
C     THE HIGHEST, IS IZONE(K,2).  IF THERE ARE NO TARGETS IN A ZONE, THE
C     IZONE VALUES ARE BOTH EQUAL TO THE INDEX NUMBER OF THE LAST TARGET
C     AS IZONE(K-1,2)).
        DO 10 I=1,NT
        AA = TO(I,1)/500.
   10   IZ(I) = AA
        IZONE(1,1) = 0
        IZONE(1,2) = 0
        IF (IZ(1) .NE. 0)   GO TO 14
```

```
      K = 1
      IZONE(1,1) = 1
      IZONE(1,2) = 1
      DO 12  I = 2,NT
      IF (IZ(I) .GT. 0)   GO TO 22
   12 IZONE(1,2) = I
C     ALL HITS IN ZONE #1 HERE.
      GO TO 30
   14 DO 16  K = 2,50
      IF (IZ(1) .EQ. (K-1))   GO TO 18
      IZONE(K,1) = 0
   16 IZONE(K,2) = 0
   18 CONTINUE
      IZONE(K,1) = 1
      IZONE(K-1,2) = 1
      IZONE(K,2) = 1
C     AT THIS POINT K IS ZONE OF FIRST HIT
      DO 20  I = 2,NT
      IF (IZ(I) .GT. (K-1))   GO TO 22
   20 IZONE(K,2) = I
C     ON TRANSFER TO '22' K IS FIRST OCCUPIED ZONE AND I IS FIRST HIT
C     IN (K+1) ZONE.
   22 CONTINUE
      N = I
      DO 28  I = N,NT
C     SKIP TO 26 IF HIT IN ZONE OF PRIOR HIT
      IF (IZ(I) .EQ. IZ(I-1))   GO TO 26
   24 K = K+1
      IZONE(K,1) = I-1
      IZONE(K,2) = I-1
C     IF NO HITS IN ZONE INCREMENT ZONE
      IF (IZ(I) .GT. (K-1))   GO TO 24
      IZONE(K,1) = I
      IZONE(K,2) = I
      GO TO 28
C     INCREMENT UPPER HIT IN ZONE
   26 IZONE(K,2) = I
   28 CONTINUE
   30 CONTINUE
      IF ((K+1) .GT. 50)   GO TO 36
C     FILL ALL EXCESS ZONES
      K1 = K + 1
      DO 32  L = K1,50
      IZONE(L,1) = NT + 1
   32 IZONE(L,2) = NT + 1
   36 CONTINUE
      IF (KTEST .LT. 3)   GO TO 50
      WRITE (6,101)
      DO 40  K = 1, 50
   40 WRITE (6,102)  K, IZONE(K,1),IZONE(K,2),TO(IZONE(K,1),2),
     X  TO(IZONE(K,2),2)
   50 CONTINUE
      RETURN
  101 FORMAT ('1',   'TARGETS BY ZONE',/,
     X'ZONE   LOWER  UPPER      (LOWER  UPPER)')
  102 FORMAT (' ',  I4,4X,I3,4X,I3,7X,F4.0,3X,F4.0)
      END
      SUBROUTINE BOMB
      COMMON /ARRAYS/ TGT(250,13), ATT(50,11),AMD(10,20,2), TO(250,2),
     XIZONE(50,2),NHIT(250),MHIT(20),HIT(20,3,25),NPW(5),HITR(5,3,250)
     X ,P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNREL(10),NCBU(250)
```

```
      COMMON/INT/NT,NA,ND,MTM,KTEST,MCR,MCW,MCL,MODE,NPRINT,NAM,NST,MTT
       COMMON /CBUHIT/ CBUHT(200,2), IR(50), KCBU, KPTI
       IF (KTEST .GT. 2)   WRITE (6,102)
       NCBUHT = 0
       KCBU   = 0
       DO 40 I=1,NA
       IR(I) = 0
       NW = ATT(I,9)
       EMD = AMD(NW,1,1)
       IF ((MODE .LT. 0) .OR. (ATT(I,11) .EQ. 1.0))   GO TO 3
C      IF  (MODE .EQ. 0)   GO TO 1
C      RN = RANDT(1.)
C      GO TO 2
    1  RN = RAN(1)
    2 IF (RN .LE. ATT(I,11))   GO TO 3
       IR(I) = 1
       GO TO 40
    3  CONTINUE
       IF (EMD .LT. 0.0)    KCBU = 1
       DX=0
       DY=0
       NS=ATT(I,7)
       PHI=ATT(I,1)/57.3
       S=SIN(PHI)
       C=COS(PHI)
       SIGR=1.483*ATT(I,4)
       SIGD=1.483*ATT(I,5)
       CALL GAUSS(SIGR,RERR)
       CALL GAUSS(SIGD,DERR)
       AGZX=ATT(I,2)+RERR*S+DERR*C
       AGZY=ATT(I,3)+RERR*C-DERR*S
       BDGZX=AGZX-S*ATT(I,8)/2.
       BDGZY=AGZY-C*ATT(I,8)/2.
       IF (NS .LT. 2)  GO TO 10
       D=ATT(I,8)/(NS-1)
       DX=S*D
       DY=C*D
   10  CONTINUE
      SIGX = ATT(I,6)
      SIGY = ATT(I,10)
       DO 20 M=1,NS
       IF (EMD .LT. 0.0)   NCBUHT = NCBUHT + 1
       IF (NCBUHT .GT. 200)   GO TO 60
      IF (MODE .LT. 0)   GO TO 13
C     IF (MODE .EQ. 0)   GO TO 11
C     RN = RANDT(1.)
C     GO TO 12
   11 RN = RAN(1)
   12  IF (RN .GT. WPNREL(NW))   GO TO 17
   13 CONTINUE
      CALL GAUSS(SIGX,X)
      CALL GAUSS(SIGY,Y)
       BAGZX=BDGZX+X
       BAGZY=BDGZY+Y
       IF ((KTEST .GT. 2) .OR. (NPRINT .LT. -1))
     X     WRITE (6,101) I, M, BAGZX, BAGZY
       IF (EMD .GE. 0.0)   GO TO 16
       CBUHT(NCBUHT,1) = BAGZX
       CBUHT(NCBUHT,2) = BAGZY
       GO TO 18
   16  IF ((BAGZX+BAGZY) .LT. -500.)   GO TO 18
```

```
      INDEX = I
      CALL TESTHT(INDEX,BAGZX,BAGZY)
      GO TO 18
 17   IF (EMD .LT. 0.0)   CBUHT(NCBUHT,1) = -10000.
 18   BDGZX=BDGZX+DX
      BDGZY=BDGZY+DY
 20   CONTINUE
 40   CONTINUE
      RETURN
 60   WRITE(6,103)
      STOP
101   FORMAT(' ',/,'ATTACK #',I4,'   BOMB #',I4,'   X-DIM ',F8.0,
     X   '  Y-DIM  ', F8.0)
102   FORMAT ('1',30('*'),'  BOMB IMPACT AND HIT DATA  ',30('*'),/,/)
103   FORMAT('0','THE CBUHT ARRAY MUST BE ENLARGED TO ',
     X   'ACCOMODATE MORE CBU WEAPONS')
      END
      SUBROUTINE  TESTHT(I,BX,BY)
      COMMON /ARRAYS/ TGT(250,13), ATT(50,11),AMD(10,20,2), TO(250,2),
     XIZONE(50.2),NHIT(250),MHIT(20),HIT(20,3,25),NRW(5),HITR(5,3,250)
     X ,P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNREL(10),NCBU(250)
     COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,MCH,MCL,MODE,NPRINT,NAM,NST,MTT
      XY = BX + BY
      NN=XY/500.
      K = MAXO(NN+1,1)
      K = MINO(K,49)
      LL = 0
      IF (K .EQ. 1)  GO TO 10
      LL = IZONE (K-1, 1)
      IF (K .EQ.2)  GO TO 20
      IF ((IZONE(K-1,1) .EQ. IZONE(K-2,2)) .AND.
     X (IZONE(K-2,2) .NE. 1))    LL = IZONE(K,1)
      GO TO 20
 10   LL = IZONE (1, 1)
 20   CONTINUE
      LU = IZONE ((K+1), 2)
      IF (KTEST .GT. 3)   WRITE (6,102)  I, LL,LU
      DO 100 IL = LL, LU
C    CONSIDER ALL TARGETS BETWEEN THE LIMITS OF LL AND LU
      L = TO(IL,2)
     IF (L .LE. 0)   GO TO 100
      IF ((TGT(L,12) + TGT(L,13)) .GT. 500.)   GO TO 100
      D = 1.414*AMD(ATT(I,9),TGT(L,10),1)
      IF ((TGT(L,1) - D) .GT. BX)   GO TO 100
      IF ((TGT(L,4) + D) .LT. BY)   GO TO 100
      IF ((TGT(L,5) + D) .LT. BX)   GO TO 100
      IF ((TGT(L,8) - D) .GT. BY)   GO TO 100
      IF (KTEST .GT. 4) WRITE (6,101)I,L,BX,BY
      CALL HITTGT(I,L,BX,BY)
100   CONTINUE
      DO 120   L = 1, NT
      IF ((TGT(L,12) + TGT(L,13)) .LE. 500.)   GO TO 120
      IF (KTEST .GT. 4)   WRITE (6,101) I,L,BX,BY
      IL = L
      CALL HITTGT(I,IL,BX,BY)
120   CONTINUE
130   CONTINUE
      RETURN
101   FORMAT (' ',10X,'TESTHIT:  ATTACK ',I3,'     TGT ',I4,  '   X-DIM
     X ', F6.0, '   Y-DIM ', F6.0)
102   FORMAT(' ',20X,'ATTACKER ',I3,'  TARGET RANK LIMITS ',2I6,/)
```

```
      END
      SUBROUTINE HITTGT(I,L,BX,BY)
      COMMON /ARRAYS/ TGT(250,13), ATT(50,11),AMD(10,20,2), TO(250,2),
     XIZONE(50,2),NHIT(250),MHIT(20),HIT(20,3,25),NRW(5),HITR(5,3,250)
     X ,P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNREL(10),NCBU(250)
      COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,MCW,MCL,MODE,NPRINT,NAM,NST,MTT
      DIMENSION DN(4),F(4),IS(4),D2(4)
      D = 0
      T = 0
      DD = AMD(ATT(I,9),TGT(L,10),1)
      S = SIN (TGT (L,9)/57.3)
      C = COS (TGT (L,9)/57.3)
      CO= C
      IF (DD .EQ. 0.0) GO TO 10
      T = (S + C)*DD
      D = (S - C)*DD
   10 X1 = TGT (L,1) - T
      Y1 = TGT (L,2) + D
      X2 = TGT (L,3) + D
      Y2 = TGT (L,4) + T
      X3 = TGT (L,5) + T
      Y3 = TGT (L,6) - D
      X4 = TGT (L,7) - D
      Y4 = TGT (L,8) - T
     IF ((BX .LT. X1) .OR. (BX .GT. X3)) GO TO 100
     IF ((BY .LT. Y4) .OR. (BY .GT. Y2)) GO TO 100
      IF (TGT (L, 9) .EQ. 0.) GO TO 60
      T = S/C
      C = 1./T
      IF ((BX .LT. X2) .AND. (BY .GT. (Y1+C*(BX-X1))))    GO TO 100
      IF ((BX .GT. X2) .AND. (BY .GT. (Y2-T*(BX-X2))))    GO TO 100
      IF ((BX .GT. X4) .AND. (BY .LT. (Y4+C*(BX-X4))))    GO TO 100
      IF ((BX .LT. X4) .AND. (BY .LT. (Y1-T*(BX-X1))))    GO TO 100
   60 CONTINUE
      IF ((KTEST .GT. 2) .OR. (NPRINT .LT. 0)) WRITE(6,101)L,BX,BY
      NHIT(L) = NHIT(L) + 1
      NW = ATT(I,9)
      NTGT = TGT(L,10)
      IF ((NTGT .EQ. 1) .AND. (MCR .NE. 0))   GO TO 110
      PP = AMD(NW,NTGT,2)
C   RESULTS INCLUDE AN ESTIMATE OF THAT FRACTION OF THE TARGET AREA THAT
C   IS COVERED BY A CIRCLE OF RADIUS 'EMD'.  IF A VALUE, NOT GREATER
C   THAN ONE, IS INPUT WITH THE SUPPLEMENTARY EMD CARD, A HIT WILL BE
C   ASSUMED TO ACHIEVE THAT FRACTIONAL KILL OF THE TARGET.  IF A VALUE
C   GREATER THAN ONE IS SPECIFIED, THAT VALUE WILL BE USED IN THE
C   MANNER AS THE EMD TO COMPUTE AN ESTIMATE OF THE TARGET FRACTION
C   THAT IS COVERED.
C   COMPUTE DISTANCES NORMAL TO THE FOUR SIDES OF THE TARGET.
      IF (TGT(L,9) .EQ. 0.0)   GO TO 75
      YD = Y1 + C*(BX-X1) - BY
      DN(1) = S*YD
      YD = Y2 - T*(BX-X2) - BY
      DN(2) = CO*YD
      YD = BY - Y4 - C*(BX-X4)
      DN(3) = S*YD
      YD = BY - Y1 + T*(BX - X1)
      DN(4) = CO*YD
      GO TO 80
   75 CONTINUE
      DN(1) = BX - X1
      DN(2) = Y2 - BY
```

```
      DN(3) = X3 - BX
      DN(4) = BY - Y1
   80 CONTINUE
      TOT = 0.0
      DO 83   N = 1,4
      IS(N) = 0
      D2(N) = DD-DN(N)
      IF (D2(N) .LE. 0.0)   GO TO 83
      TOT = TOT + D2(N)*D2(N)
   83 CONTINUE
      IF (TOT .GT. DD*DD)   GO TO 100
      RL = DD
      IRAD = 0
   81 CONTINUE
      AL = 3.14159*RL*RL
      DO 84   N = 1,4
      F(N) = 0.0
      R = DD - DN(N)
      IF ((R .GT. 0.0) .AND. (R .GE. RL))   F(N) = 1.0
      IF (F(N) .EQ. 1.0)   GO TO 84
      IF ((R .LT. 0.0) .AND. (-R .GT. RL))   GO TO 84
      IF (R .LT. 0.0)   R = -R
      IF (R .EQ. 0.0)   GO TO 82
      Z = R/RL
      RHO = 2.*ATAN((1./(Z*Z)-1.)**(.5))
      A = RL*RL*(RHO - SIN(RHO))/2.
      IF (DN(N) .GT. DD)   A = AL - A
      F(N) = 1. - A/AL
      GO TO 84
   82 F(N) = .5
   84 CONTINUE
      FX = 1. - F(1) - F(3)
      FY = 1. - F(2) - F(4)
      IF (FX .LT. 0.0)   FX = 0.0
      IF (FY .LT. 0.0)   FY = 0.0
      IF (KTEST .GT.4)   WRITE (6,102)  FX, FY
C     NOTE THAT THE USE OF FX AND FY PROVIDES ONLY AN APPROXIMATE RESULT
      FAC = FX*FY*AL/(TGT(L,12)*TGT(L,13))
      PS = 1. - AMIN1(1., FAC)
      IF (IRAD .EQ. 1)   GO TO 91
   90 P(L,1) = 1. - (1.-P(L,1))*PS
      IF (PP .LE. 1)   GO TO 92
      IRAD = 1
      RL = PP
      GO TO 81
   91 P(L,3) = 1. - (1. - P(L,3))*PS
      GO TO 93
   92 IF (PP .EQ. 0.0)   GO TO 93
      P(L,3) = 1. - (1. -P(L,3))*(1. - PP)
   93 CONTINUE
      IF (KTEST .GT. 3)    WRITE (6,1001)  L, P(L,1), P(L,3)
      IF (TGT(L,11) .LT. 1.)   GO TO 100
      IF (NHIT(L) .GT. 25)   GO TO 100
      DO 95   J = 1, NST
      IF (MHIT(J) .EQ. 0)  GO TO 100
      IF (MHIT(J) .NE. L)   GO TO 95
      HIT (J,1,NHIT(L))=BX
      HIT (J,2,NHIT(L))=BY
      HIT (J,3,NHIT(L))=ATT(I,9)
      GO TO 100
   95 CONTINUE
```

```
      100   CONTINUE
            IF (NTGT .NE. 1) RETURN
      110   CONTINUE
            DO 120   J = 1,5
            IF (NRW(J) .EQ. 0)  GO TO 130
            IF (NRW(J) .NE. L)  GO TO 120
            HITR(J,1,NHIT(L)) = BX
            HITR(J,2,NHIT(L)) = BY
            HITR(J,3,NHIT(L)) = ATT(I,9)
      120   CONTINUE
      130   CONTINUE
            RETURN
      101   FORMAT(' ',15X,'***** HITTGT   NHIT(',I3,')',2F6.0)
      102   FORMAT(' ', '    FX ',F6.3,'    FY ',F6.3)
     1001   FORMAT(' ','  TGT # ',I3,'    PK1 ',F8.3,'    PK3 ',F8.3)
            END
            SUBROUTINE GAUSS (S,V)
           COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,MCW,MCL,MODE,NPRINT,NAM,NST,MTT
            IF ((KTEST .GT. 7) .OR. (MODE .LT. 0))   GO TO 50
            A=0.0
C           IF (MODE .EQ. 0)   GO TO 20
C           DO 10 I=1,12
C           Y = RANDT(1.)
C 10        A = A + Y
C           GO TO 40
       20   DO 30 I=1,12
            Y=RAN(1)
       30   A = A + Y
       40   V = (A-6.0)*S
            RETURN
       50   CONTINUE
            V = 0.0
            RETURN
            END
            SUBROUTINE CHECKR
            COMMON /ARRAYS/ TGT(250,13), ATT(50,11),AMD(10,20,2), TD(250,2),
          X IZONE(50,2),NHIT(250),MHIT(20),HIT(20,3,25),NRW(5),HITR(5,3,250)
          X ,P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNREL(10),NCBU(250)
           COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,MCW,MCL,MODE,NPRINT,NAM,NST,MTT
            COMMON/STATS/NTRIAL,ITRIAL,NSTAT,STAT(250,8),STAT2(20,5),MSTAT(8)
            COMMON /CONTRL/ NREP,NPLOT,INW,INL,NSAVE1,NSAVE2,LIST,NJMEM
            NC=0
            NN=0
            LHOLES = 10000
            DO 40 MRW = 1, 5
C         CYCLE THRU AS MANY AS 8 RUNWAY/TAXIWAYS.
            IRW = NRW(MRW)
C         EXIT IF NO TARGET NUMBER (IRW) FOUND.
            IF (IRW .EQ. 0)  GO TO 50
            NN=NN+1
            IF (NHIT(IRW) .EQ. 0)   GO TO 40
            IF (KTEST .GT. 4)   WRITE (6,102) IRW
             INDEX = MRW
             CALL RUNWAY (INDEX, IRW, ICOND, NHOLES)
C         RUNWAY SUBROUTINE RETURNS ICOND = 0 IF RUNWAY HAS REQUIRED SPACE;
C            ICOND = 1 IF NOT.
C
            IF (ICOND .EQ. 1) NC=NC+1
            IF (NHOLES .LT. LHOLES)   LHOLES = NHOLES
            IF (NPRINT .EQ. 4)   WRITE(6,103)ITRIAL,IRW,NHIT(IRW),NHOLES
       40   CONTINUE
```

```
   50 IF (NC .EQ.NN)   GO TO 60
      IF (NPRINT .LT. 3)   WRITE (6,101)
      NSTAT = NSTAT + 1
   60 CONTINUE
      IF ((NREP .EQ. 1) .AND. (NC .EQ. NN))   GO TO 70
      RETURN
   70 MSTAT(1) = MSTAT(1) + LHOLES
      MSTAT(2) = MSTAT(2) + LHOLES*LHOLES
      RETURN
  101 FORMAT(' ',' AT LEAST ONE RUNWAY IS AVAILABLE')
  102 FORMAT(' ',' CHECK TARGET #', I5)
  103 FORMAT(' ','TRIAL',I4,'   TGT',I4,'   HITS',I4,'   REPAIRS',I3)
      END
      SUBROUTINE RUNWAY (MRW, IRW, ICOND, NHOLES)
      COMMON /ARRAYS/ TGT(250,13), ATT(50,11),AMD(10,20,2), TO(250,2),
    X IZONE(50,2),NHIT(250),MHIT(20),HIT(20,3,25),NRW(5),HITR(5,3,250)
    X ,P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNKEL(10),NCBU(250)
      COMMON/STATS/NTRIAL,ITRIAL,NSTAT,STAT(250,8),STAT2(20,5),MSTAT(8)
     COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,MCW,MCL,MODE,NPRINT,NAM,NST,MTT
       COMMON /HITS/ XN(250),YN(250),NZ(250)
      COMMON /CONTRL/ NREP,NPLOT,INW,INL,NSAVE1,NSAVE2,LIST,NJMEM
      DIMENSION NTEST(250), YH(250,2)
C
C    CHECKS FOR THE EXISTANCE OF THE SPECIFIED RUNWAY MINIMUMS (MCL X
C    MCW) ON EACH RUNWAY AND DESIGNATED TAXIWAY (TYPE #1 TARGETS).
C    STOPS SEARCHING A GIVEN RUNWAY WHENEVER REQUIREMENT IS SATISFIED.
C
      TH=TGT(IRW,9)/57.3
C
C    ESTABLISH ORIGIN (XO, YO) FOR A RECTANGULAR COORDINATE SYSTEM WITH
C    THE X-AXIS ON THE MORE SOUTHERLY EDGE OF THE RUNWAY.
C
      NHI = 0
      NHOLES =1000
      DO 5   N = 1, 250
    5 NZ(N) = 0
       IF (TGT(IRW,12) .GT. TGT(IRW,13))   GO TO 10
      NDIR=1
      XO=TGT(IRW,1)
      YO=TGT(IRW,2)
      LTH = TGT(IRW,13)
      WID = TGT(IRW,12)
      GO TO 20
   10 NDIR=2
      XO=TGT(IRW,7)
      YO=TGT(IRW,8)
      LTH = TGT(IRW,12)
      WID = TGT(IRW,13)
   20 CONTINUE
     IF (KTEST .GT. 4)  WRITE (6,1004)IRW,XO,YO,LTH,WID,MCL,MCW
      IF (MCW .GT. WID)   GO TO 320
      NHIT1 = NHIT(IRW)
      DO 50   I = 1, NHIT1
      IF (AMD(HITR(MRW,3,I),1,1) .LT. 0.)   GO TO 50
      NTW = HITR(MRW,3,I)
      GO TO 60
   50 CONTINUE
   60 CONTINUE
      NON = 1
      EMD = AMD(NTW,1,1)
      DO 70   I = 1, NHIT1
```

```
          IF (HITR(MRW,3,I) .EQ. NTW)   GO TO 70
          IF (AMD(HITR(MRW,3,I),1,1) .LT. 0.)   GO TO 70
          NON = 0
          EMD = 0.0
          GO TO 80
   70     CONTINUE
   80     CONTINUE
          DO 140   I = 1, NHIT1
C       TRANSFORM HIT COORDINATES TO RUNWAY COORDINATES.
          NHI = NHI + 1
          XB = HITR(MRW,1,I)
          YB = HITR(MRW,2,I)
          IF (AMD(HITR(MRW,3,I),1,1) .LT. 0.)   NZ(I) = 1
          IF (TH .EQ. 0.0)  GO TO 110
          XX=XB-XO
          YY=YB-YO
          R = (XX*XX+YY*YY)**(0.5)
          IF (KTEST .GT. 7)  WRITE (6,1010) R,XX,YY
          YZ = YY/R
         TH1 = ATAN(YY/XX)
          IF (XX .LT. 0.0)   TH1 = TH1 + 3.1416
         TH2 = TH1 + TH
          IF (NDIR .EQ. 2)   TH2 = TH2 - 1.5708
          XN(I)=R*COS(TH2)
          YN(I)=R*SIN(TH2)
          IF (KTEST .GT. 5)   WRITE(6,1009)I,XN(I),YN(I)
          GO TO 130
  110     IF (NDIR .EQ. 2)  GO TO 120
          XN(I)=XB-XO
          YN(I)=YB-YO
          GO TO 130
  120     XN(I)=YB-YO
          YN(I)=XO-XB
  130     IF ( I .GT. 249)   GO TO 150
  140     CONTINUE
          GO TO 160
  150     WRITE (6,1001) IRW, ITRIAL
  160     CONTINUE
          NH = NHI
C       IF NPLOT .EQ. 2 RUNWAY IMPACTS ARE PLOTTED FOR ALL CONDITIONS.
C       IF NPLOT .EQ. 1 IMPACTS ONLY PLOTTED WHEN RUNWAY IS CLOSED.
         IF ((MRW .EQ. 1) .AND. (NPLOT .EQ. 2))  WRITE(6,1012)ITRIAL
          IF (NPLOT .EQ. 2)   CALL PLOTHT(NH,IRW,LTH,WID)
          IF (KTEST .GT.6)   WRITE (6,1006) NHIT(IRW),NH
          DO 170 I=1,NH
          YH(I,1) = YN(I)
  170     YH(I,2)=I
          IF (NH .EQ. 1)  GO TO 190
          DO 180 J=2,NH
          DO 180 K=2,NH
C       ORDER ALL HITS FROM LOWEST Y TO HIGHEST.  YH(I,1) IS THE
C       Y COORDINATE, YH(I,2) THE HIT NUMBER, OF THE I TH ORDERED HIT.
          I=NH-K+2
          IF (YH(I,1) .GE. YH(I-1,1))  GO TO 180
          T=YH(I-1,1)
          TN=YH(I-1,2)
          YH(I-1,1)=YH(I,1)
          YH(I-1,2)=YH(I,2)
          YH(I,1)=T
          YH(I,2)=TN
  180     CONTINUE
```

```
  190   CONTINUE
        XL = 0.0
        XU = MCL
  200   YL = 0.0
        YU = MCW
        NYL = 1
        IF (NON .EQ. 0)   GO TO 210
        YL = YL - EMD
        YU = YU + EMD
  210   CONTINUE
  220   IHOLE = 0
        DO 250   I = NYL , NH
        IF (NZ(I) .EQ. 1)   GO TO 250
        YT = YH(I,1)
        IF (NON .EQ. 1)   GO TO 230
        R = AMD(HITR(MRW,3,YH(I,2)),1,1)
        YL = YL - R
        YU = YU + R
  230   IF (YT .LT. YL)   GO TO 240
        XT = XN(YH(I,2))
        IF ((XT .LT. XL) .OR. (XT .GT. XU))   GO TO 240
        IF (YT .GT. YU)   GO TO 260
        IHOLE = IHOLE + 1
        IF (KTEST .GT. 6)   WRITE(6,1008)  I, IHOLE, YL,YU
        NTEST(IHOLE) = I
        IF (NREP .EQ. 0)   GO TO 260
  240   IF (NON  .EQ. 1)   GO TO 250
        IF (I .EQ. NH)   GO TO 260
        YL = YL + R
        YU = YU - R
  250   CONTINUE
  260   CONTINUE
        IF (IHOLE .EQ. 0)   GO TO 300
        IF (NON .EQ. 1)   NYL = NTEST(1)
        NHOLES = MINO(NHOLES, IHOLE)
        IF (NON .EQ. 0)   GO TO 270
        YL = YL + INW
        YU = YU + INW
        IF (YU .GT. (WID+EMD))   GO TO 280
        GO TO 220
  270   YL = YL - R + INW
        YU = YU - R + INW
        IF (YU .GT. WID)   GO TO 280
         IF (KTEST .GT. 4)   WRITE(6,1008) NHOLES
        GO TO 220
  280   CONTINUE
        XL = XL + INL
        XU = YU + INL
        IF (XU .GT. LTH)   GO TO 290
        GO TO 200
  290   IF (NPRINT .GE. 3)   GO TO 295
        IF ((MRW .EQ. 1) .AND. (NPLOT .NE. 2))  WRITE (6,1012)ITRIAL
         WRITE(6,1002) IRW
        IF (NREP .GT. 0)   WRITE (6,1011) NHOLES
        IF (NPLOT .EQ. 1)   CALL PLOTHT(NH,IRW,LTH,WID)
  295   ICOND = 1
        IF ((NPLOT .GT. 0) .AND. (NPRINT .LT. 3)) WRITE(6,1000)
         RETURN
  300   IF (NPRINT .GT. 2)   GO TO 310
        IF ((MRW .EQ. 1).AND.(NPLOT .NE. 2))   WRITE (6,1012) ITRIAL
        WRITE (6,1003) IRW
```

```
      310   ICOND = 0
            IF ((NPLOT .EQ. 2) .AND. (NPRINT .LT. 3)) WRITE(6,1000)
            NHOLES = 0
            RETURN
      320   WRITE(6,1005)  IRW
            STOP
      1000   FORMAT('1')
      1001   FORMAT('0','ONLY FIRST 250 HITS TESTED FOR TARGET #',I4,
          X  ' IN TRIAL #',I4,/,/)
      1002   FORMAT('0','RUNWAY #',I3,' IS CLOSED',/)
      1003   FORMAT('0','RUNWAY #',I3,' IS OPEN',/)
      1004   FORMAT(' ','  RUNWAY SPECS ',I4,4X,2F8.0,I8,F8.0,2I8)
      1005   FORMAT ('0','***** TARGET #',I3,' IS TOO NARROW FOR ',
          X  'FLIGHT OPERATIONS')
      1006   FORMAT(' ','   # HITS TO CHECK', 2I6)
      1008   FORMAT(' ','   TEST POINT #B ',2I10,2F10.0)
      1009   FORMAT(' ','   TEST POINT #E', I4,2F10.0)
      1010   FORMAT(' ','    R = ',F8.0,10X,2F10.0)
      1011   FORMAT(' ', I4,' HOLES MUST BE REPAIRED TO MEET RUNWAY ',
          X  'MINIMUMS',/)
      1012   FORMAT ('1', '     *****  TRIAL #',I3,'  *****',/)
            END
            SUBROUTINE PRINT
            INTEGER *4 NAME
            COMMON /ARRAYS/ TGT(250,13), ATT(50,11),AMC(10,20,2), TO(250,2),
           XIZONE(50,2),NHIT(250),MHIT(20),HIT(20,3,25),NKW(5),HITR(5,3,250)
           X ,P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNREL(10),NCBU(250)
            COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,MCW,MCL,MODE,NPRINT,NAM,NST,MTT
            COMMON/STATS/NTRIAL,ITRIAL,NSTAT,STAT(250,8),STAT2(20,5),MSTAT(8)
            IF (NTRIAL .EQ. 1)   GO TO 1
            WRITE (6,108)   ITRIAL
            GO TO 3
        1   CONTINUE
            WRITE (6,106)
        3   CONTINUE
            WRITE (6,101)
            DO 10 M=1,MTT
            NN= 0
            DO 10  I=1,NT
            IF (TGT(I,10) .NE. M)  GO TO 10
            NN= NN+1
            IF (NN .EQ. 1)   WRITE (6,109) M
            NAID = NHIT(I) - NCBU(I)
            WRITE(6,102) I, NAID,COV(I),P(I,1),P(I,3),P(I,2),
          X   NAME(I,1), NAME(I,2)
       10   CONTINUE
            IF (NPRINT .GT. 0)   GO TO 30
            WRITE (6,103)
             DO 20   M = 1, NST
            IF (MHIT(M) .EQ. 0)   GO TO 30
            NN=0
            NL = NHIT(MHIT(M))
            IF (NL .EQ. 0)   GO TO 20
            DO 15   N=1, NL
            NN=NN+1
            IF (NN .EQ. 26)   GO TO 20
            IF (NN .EQ. 1) WRITE (6,104)  MHIT(M)
            X=HIT(M,1,NN)
            Y=HIT(M,2,NN)
            NWPN=HIT(M,3,NN)
            WRITE (6,105) X,Y,NWPN
```

```
   15   CONTINUE
   20   CONTINUE
   30   CONTINUE
        DO 50   M = 1, 5
        IF (NRW(M) .EQ. 0)   GO TO 60
        NN = 0
        NL = NHIT(NRW(M))
        IF (NL .EQ. 0)   GO TO 50
        DO 40   N = 1, NL
        NN = NN + 1
        IF (NN .EQ. 251)   GO TO 50
        IF (NN .EQ. 1)   WRITE (6,104)   NRW(M)
        X = HITR(M,1,NN)
        Y = HITR(M,2,NN)
        NWPN = HITR(M,3,NN)
        WRITE (6,105) X, Y, NWPN
   40   CONTINUE
   50   CONTINUE
   60   CONTINUE
        RETURN
  101   FORMAT('0',
      X'TGT    NO.       CBU      ',10X,
      X'  BOMBS            CBU           BLDG',/,1X,
      X'NO.   HITS    COVERAGE   ',10X,
      X'EMD   OTHER        PK            NO.')
  102   FORMAT (' ',I3,3X,I4,4X,F6.2,10X,2(3X,F5.3),6X,F5.3,7X,2A4)
  103   FORMAT('0',15X,'HIT LOCATION AND WPN TYPE FOR SELECTED TARGETS'
      X,/,/)
  104    FORMAT(' ','   TARGET NUMBER',I4,'   X-DIM    Y-DIM   ',
      X 'WPN TYPE',/)
  105    FORMAT(' ',  21X,2F9.0,I7)
  106    FORMAT('1',20X,'TARGET HIT SUMMARY',/)
  108    FORMAT('1',10X,'TARGET HIT SUMMARY       TRIAL',I5)
  109   FORMAT('0', 10X,'** TARGET TYPE # ',I3,'  **',/)
        END
        SUBROUTINE STATIS
        COMMON /ARRAYS/ TGT(250,13), ATT(50,11),AMD(10,20,2), TO(250,2),
      XIZONE(50,2),NHIT(250),MHIT(20),HIT(20,3,25),NRW(5),HITR(5,3,250)
      X ,P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNREL(10),NCBU(250)
       COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,NCW,MCL,MODE,NPRINT,NAM,NST,MTT
        COMMON/STATS/NTRIAL,ITRIAL,NSTAT,STAT(250,8),STAT2(20,5),MSTAT(8)
         COMMON /CONTRL/ NREP,NPLOT,INW,INL,NSAVE1,NSAVE2,LIST,NJMEM
        WRITE (6,101)  NTRIAL
        AVGREP = 0.0
        DO 12   M = 1, MTT
        NN = 0
        SUM1 = 0.0
        SUM2 = 0.0
        DO 10   I = 1, NT
        IF (TGT(I,10) .NE. M)   GO TO 10
        CONTINUE
        NN = NN + 1
        IF (NN .EQ. 1)   WRITE (6,102) M
        FHIT = (STAT(I,3)/NTRIAL)*100.
        AHITS=  STAT(I,1)/NTRIAL
        SUM1 = SUM1 + AHITS
        TRIAL = NTRIAL
        SDH = (STAT(I,2) - TRIAL*AHITS*AHITS)/(TRIAL - 1.)
        SDH = SDH**(0.5)
        ACOV = STAT(I,4)/TRIAL
        SUM2 = SUM2 + ACOV
```

```
          SDC  = (STAT(I,5) - TRIAL*ACOV*ACOV)/(TRIAL - 1.)
          SDC = SDC**(0.5)
          PK1  = STAT(I,6)/TRIAL
          PK2  = STAT(I,7)/TRIAL
          PK3  = STAT(I,8)/TRIAL
          WRITE (6,103) I,FHIT,AHITS,SDH,ACOV,SDC,PK1,PK3,PK2,NAME(I,1),
         X  NAME(I,2)
   10     CONTINUE
          IF (NN .GT. 0)   WRITE (6,108)   SUM1, SUM2
   12     CONTINUE
          WRITE (6,106)
          DO 25    M = 1,MTT
          DO 15    I = 1,NT
          IF ( TGT(I,10) .EQ. M)    GO TO 20
   15     CONTINUE
          GO TO 25
   20     FHIT = STAT2(M,1)/TRIAL
          SDH = (STAT2(M,2) - TRIAL*FHIT*FHIT)/(TRIAL - 1.)
          FCOV3 = 100.*(STAT2(M,3)/TRIAL)
          FCOV4 = 100.*(STAT2(M,4)/TRIAL)
          FCOV5 = 100.*(STAT2(M,5)/TRIAL)
          FHIT = 100.*FHIT
          SDH = 100.*(SDH**(0.5))
          WRITE (6,107) M, FHIT, SDH, FCOV3, FCOV5, FCOV4
   25     CONTINUE
   30     CONTINUE
          IF (MCF .EQ. 0)    GO TO 50
          STA = NSTAT
          FOPEN = (STA/TRIAL)*100.
          NCLSD = NTRIAL - STA
          IF (NCLSD .EQ. 0)    GO TO 40
          CLSD = NCLSD
          AVGREP = MSTAT(1)/CLSD
          SDREP  = 0.0
          IF (NCLSD .GT. 1)
         X  SDREP = ((MSTAT(2)-CLSD*AVGREP*AVGREP)/(CLSD-1.))**(.5)
   40     CONTINUE
          WRITE (6,104)  FOPEN
          IF (NREP .EQ. 1)    WRITE (6,105) AVGREP, SDREP
   50     CONTINUE
          RETURN
  101     FORMAT ('1', 10X,'TARGET DAMAGE STATISTICS FOR',I4,' TRIALS',/,/,
         X  '  TARGET    PERCENT    AVERAGE HITS  STD. DEV.   AVG. CBU STD.'
         X,' DEV.   AVG. BOMB COVERAGE  CBU     BLDG',/,
         X  '  NUMBER  ATTACKS HIT  PER ATTACK    OF HITS   COVERAGE COVE',
         X'RAGE       EMD     OTHER          PK        NO.',/,/)
  102     FORMAT (' ',/, 15X, 'TARGET TYPE # ',I3,/)
  103     FORMAT(' ',I6,6X,F6.1,6X,F7.2,6X,F6.2,4X,F6.2,4X,F6.2,
         X   2F9.3,3X,F9.3,2X,2A4)
  104     FORMAT(' ',/, ' AT LEAST ONE MINIMUM RUNWAY SECTION WAS OPEN AFTE
         XR', F6.1,' PERCENT OF THE ATTACKS',/)
  105     FORMAT(' ','WHEN ALL RUNWAYS WERE CLOSED, ',F4.1,'(',F4.1,
         X') HOLES REQUIRED REPAIR, ON THE AVERAGE, TO PROVIDE',
         X  ' A MINIMUM RUNWAY',/)
  106     FORMAT(' ',/,/,10X,'DAMAGE STATISTICS BY TARGET TYPE',/,/,
         X  '                  AVERAGE',/,
         X  '   TARGET    PERCENT   STANDARD    ---- COVERAGE ----',/,
         X  '    TYPE        HIT     DEVIATION    EMD    OTHER      CBU'/)
  107     FORMAT(' ',5X, I2, 2(6X,F5.1), 3(2X,F5.1))
  108    FORMAT(' ',25X,'------',16X,'------',/,26X,F6.2,16X,F6.2)
          END
```

```
      SUBROUTINE CBU
      COMMON /ARRAYS/ TGT(250,13), ATT(50,11),AM[illegible],20,2), TD(250,2),
     XIZONE(50,2),NHIT(250),MHIT(20),HIT(20,3,25),MRW(5),HITR(5,3,250)
     X ,P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNREL(10),NCBU(250)
      COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,MCW,MCL,MODE,NPRINT,NAM,NST,MTT
      COMMON/STATS/NTRIAL,ITRIAL,NSTAT,STAT(250,8),STAT2(20,5),MSTAT(8)
      COMMON /CBUHIT/ CBUHT(200,2), IR(50), NCBU, KPTI
      DIMENSION NCOV(16,16),ICOV(16,16),PSCOV(16,16),
     X   X(4), Y(4), XX(4), YY(4), WD(10)
C     COMPUTE PATTERN DIAGONAL DIMENSION
      DO 2   NWPN = 1, 10
      WD(NWPN) = 0
      LTH = -AMD(NWPN,1,1)
      IF (LTH .LE. 0)   GO TO 2
      WID = AMD(NWPN,2,1)
      WD(NWPN) = ((LTH*LTH + WID*WID)**(.5))/2.
    2 CONTINUE
C   AT THIS POINT THE PROGRAM NOW CHECKS, TARGET BY TARGET, FOR
C   WHATEVER CBU'S MAY HAVE COVERED ANY PART OF THE TARGET.
          DO 200    L = 1, NT
C         ******************       FOR EACH TARGET
      INIT1 = 0
C    COMPUTE TARGET CENTER
      TCX=(TGT(L,1)+TGT(L,5))*0.5
      TCY=(TGT(L,2)+TGT(L,6))*0.5
C     TARGET DIAGONAL
      XA=TGT(L,1)-TGT(L,5)
      YA=TGT(L,2)-TGT(L,6)
      TD = 0.5*((XA*XA + YA*YA)**(0.5))
      NCBUHT = 0
          DO 40     I = 1, NA
C         ******************       FOR EACH ATTACK
      IF (IR(I) .EQ. 1)   GO TO 40
      INIT2 = 0
      NWPN = ATT(I,9)
      LTH = -AMD(NWPN,1,1)
      IF (LTH .LE. 0)   GO TO 40
      WID = AMD(NWPN,2,1)
      TOT = WD(NWPN) + TD
      NS = ATT(I,7)
      INIT3 = 0
          DO 20     M = 1, NS
C         ******************       FOR EACH WEAPON
      NCBUHT = NCBUHT + 1
      XB = CBUHT(NCBUHT, 1)
      IF (XB .EQ. -10000.)     GO TO 20
      YB = CBUHT(NCBUHT, 2)
C    DISTANCE BETWEEN TARGET AND PATTERN CENTERS
      D = ((XB-TCX)*(XB-TCX) + (YB-TCY)*(YB-TCY))**(.5)
C    TARGET CANNOT BE HIT IF D GREATER THAN TOT
      IF (D .GT. TOT)   GO TO 20
      IF (INIT3 .GT. 0)   GO TO 16
      INIT3 = 1
      PHI = ATT(I,1)/57.3
      S = SIN(PHI)
      C = COS(PHI)
      SL = S*LTH
      SW = S*WID
      CL = C*LTH
      CW = C*WID
      DO 5   J = 1, 16
```

```
      DO 5   K = 1, 16
    5 NCOV(J,K) = 0
   16 CONTINUE
      XX(1) = XB - (SL+CW)/2.
      XX(2) = XX(1) +SL
      XX(3) = XX(2) + CW
      XX(4) = XX(1) + CW
      YY(1) = YB + (SW-CL)/2.
      YY(2) = YY(1) + CL
      YY(3) = YY(2) - SW
      YY(4) = YY(1) - SW
      IF (INIT2 .GT. 0)   GO TO 18
      INIT2 = 1
C    FIND WESTERLY CORNER
       ILX = 1
      DO 15   NN = 1,4
   15 IF (XX(NN) .LT. XX(ILX))   ILX = NN
      IF (S .EQ. 1.0)   ILX = 4
C    RENUMBER CORNERS SO THAT CORNER #1 IS THE MOST WESTERN
       IDIF = ILX - 1
C    COMPUTE AND ADJUST TAN AND COTAN AS REQUIRED
      IF ((S .EQ. 0.0) .OR. (C .EQ. 0.0))   GO TO 28
      IF ((ILX .EQ. 2) .OR. (ILX .EQ. 4))   GO TO 26
      T = S/C
      GO TO 27
   26 T = -C/S
   27 CT= 1./T
   28 CONTINUE
   18 DO 22   NN = 1,4
       NEW = NN - IDIF
      IF (NEW .LT. 1)   NEW = NEW + 4
       X(NEW) = XX(NN)
       Y(NEW) = YY(NN)
   22  IF (KTEST .GT. 4)  WRITE(6,1003)NEW,X(NEW),Y(NEW)
       IF (INIT1 .GT. 0)   GO TO 31
C    CREATE A 16-POINT GRID ON TARGET - USE MORE POINTS FOR LARGE TGTS
      INIT1 = 1
      NXO = 8
      NYO = 8
      IF (TGT(L,12) .GT. 250.)   NYO = 16
      IF (TGT(L,12) .GT. 1000.)   NYO = 32
      IF (TGT(L,13) .GT. 250.)   NXO = 16
      IF (TGT(L,13) .GT. 1000.)   NXO = 32
      NXT = NXO/2
      NYT = NYO/2
      DO 29   J = 1, NXT
      DO 29   K = 1, NYT
      ICOV(J,K) = 0
   29 PSCOV(J,K) - 1.0
       NX1 = NXO - 1
       NY1 = NYO - 1
        XO = NXO
        YO = NYO
        AO = TGT(L,1)
        A1 = (TGT(L,7)-TGT(L,1))/XO
        A2 = (TGT(L,3)-TGT(L,1))/YO
        A3 = TGT(L,2)
        A4 = (TGT(L,4)-TGT(L,2))/YO
        A5 = (TGT(L,8)-TGT(L,2))/XO
   31 CONTINUE
C    TEST TO SEE IF TARGET CORNERS COVERED BY PATTERN
```

```
          NIN = 0
          NGIN = 0
            DO 10    NC = 1, 7, 2
          XT=TGT(L,NC)
          YT=TGT(L,NC+1)
          IF (KTEST [illegible]. 4)  WRITE(6,1001) XT,YT
          IF ((X[illegible] Y(1)).OR.(XT .GT. X(3)))   GO TO 10
          IF ([illegible] Y(4)).OR.(YT .GT. Y(2)))   GO TO 10
           IF [illegible]) .OR. (C .EQ. 0.))   GO TO 9
          IF [illegible] (Y(1)+CT*(XT-X(1))))) GO TO 10
          IF [illegible].(Y(2)-T*(XT-X(2)))) GO TO 10
          IF (YT.LT.(Y(4)+CT*(XT-X(4)))) GO TO 10
          IF (YT.LT.(Y(1)-T*(XT-X(1)))) GO TO 10
    9     NIN = NIN + 1
   10     CONTINUE
          IF (KTEST .GT. 3)   WRITE(6,1002)  NIN
          IF (NIN .LT. 4)   GO TO 34
C       IF ALL CORNERS COVERED BY PATTERN, TARGET FULLY COVERED
          DO 32    J = 1, NXT
          DO 32    K = 1, NYT
   32     NCOV(J,K) = NCOV(J,K) + 1
          GO TO 33
C       IF PARTIALLY COVERED, ESTIMATE FRACTION THAT IS COVERED
   34     CONTINUE
           DO 30 NX = 1,NX1,2
           DO 30 NY = 1,NY1,2
           J = (NX+1)/2.
           K = (NY+1)/2.
C       GRID-POINT DIMENSIONS
           XT = AC + NX*A1 + NY*A2
           YT = A3 + NY*A4 + NX*A5
C       CHECK IF WITHIN RECTANGLE ENCLOSING PATTERN THAT IS PARALLEL TO
C       AXES
         IF ((XT .LT. X(1)).OR.(XT .GT. X(3)))   GO TO 30
         IF ((YT .LT. Y(4)).OR.(YT .GT. Y(2)))   GO TO 30
          IF ((S .EQ. 1.) .OR. (C .EQ. 0.))   GO TO 35
C       CHECK IF POINT IS WITHIN ACTUAL CBU PATTERN
         IF (YT .GT.(Y(1)+CT*(XT-X(1))))   GO TO 30
         IF (YT .GT.(Y(2)- T*(XT-X(2))))   GO TO 30
         IF (YT .LT.(Y(4)+CT*(XT-X(4))))   GO TO 30
         IF (YT .LT.(Y(1)-T*(XT-X(1))))   GO TO 30
   35     NGIN = NGIN + 1
          NCOV(J,K) = NCOV(J,K) + 1
          IF (KTEST .GT. 5)  WRITE(6,1005) NX,NY,XT,YT,NGIN,NCOV(J,K)
   30     CONTINUE
   33     CONTINUE
          IF ((NIN + NGIN) .EQ. 0)   GO TO 20
C       RECORD ANY COVERAGE AS A 'HIT'
          NHIT(L) = NHIT(L) + 1
          NCBU(L) = NCBU(L) + 1
          IF (KTEST .GT. 4)  WRITE(6,1006) L, NHIT(L), NCBU(L)
          IF ((TGT(L,11) .LT. 1) .OR. (TGT(L,10) .EQ. 1.))   GO TO 130
          IF (NHIT(L) .GT. 25)   GO TO 130
          DO 120    J = 1, NST
          IF (MHIT(J) .EQ. 0) GO TO 130
          IF (MHIT(J) .NE. L) GO TO 120
          HIT (J,1,NHIT(L))=XB
          HIT (J,2,NHIT(L))=YB
          HIT (J,3,NHIT(L))=NWPN
          IF (NHIT(L) .EQ. 25)   WRITE (6,1007)  L, ITRIAL
          GO TO 130
```

```
      120  CONTINUE
      130  CONTINUE
           IF (TGT(L,10) .NE. 1.)   GO TO 150
           IF (NHIT(L) .GT. 250)   GO TO 150
           DO 140   J = 1, 5
           IF (NRW(J) .EQ. 0)   GO TO 150
           IF (NRW(J) .NE. L)   GO TO 140
           HITR(J,1,NHIT(L)) = XB
           HITR(J,2,NHIT(L)) = YB
           HITR(J,3,NHIT(L)) = NWPN
           IF (NHIT(L) .EQ. 250)   WRITE(6,1009)   L, ITRIAL
      140  CONTINUE
      150  CONTINUE
C    *************************     RECYCLE FOR MORE WEAPONS
       20   CONTINUE
           IF (INIT2 .EQ. 0)   GO TO 40
           PSP = 1. - AMOD(NWPN,TGT(L,10),2)
           DO 160    J = 1, NXT
           DO 160    K = 1, NYT
           ICOV(J,K) = ICOV(J,K) + NCOV(J,K)
      160  PSCOV(J,K) = PSCOV(J,K)*PSP**NCOV(J,K)
C    *************************     RECYCLE FOR MORE ATTACKS
       40  CONTINUE
           IF ((INIT1+INIT2) .GT. 0)   GO TO 165
           P(L,2) = 0.0
           COV(L) = 0.0
           GO TO 200
      165  TCOV = 0.0
           PST = 0.0
           DO 170    J = 1, NXT
           DO 170    K = 1, NYT
           IF (ICOV(J,K) .GT. 0)   TCOV = TCOV + 1.
      170  PST = PST + PSCOV(J,K)
           TOTC = NXT*NYT
           COV(L) = TCOV/TOTC
           P(L,2) = 1. - PST/TOTC
           IF (KTEST .GT. 3)WRITE(6,1004)L,TCOV,TOTC,PST,COV(L),P(L,2)
C    **************************    RECYCLE FOR MORE TARGETS
      200  CONTINUE
           RETURN
     1001   FORMAT(' ',' XT   ',F8.0,'    YT   ',F8.0)
     1002    FORMAT(' ', '    NIN ',I4)
     1003    FORMAT(' ', '    NEW ',I3,2F10.0)
     1004    FORMAT(' ','TGT',I4,'   COV',F6.0,'     TOT',F6.0,'   PST',
         X  F10.4,'      COV',F10.4,'     P(L,2)',F10.4)
     1005    FORMAT(' ','    NX ',I3,'     NY ',I3,2F10.0,2I10)
     1006    FORMAT(' ',' TGT ',I4,'   NHIT',I5,'   NCBU',I5)
     1007   FORMAT('0','***** ONLY 25 HITS WERE STORED FOR TARGET #',
         X  I3,' DURING TRIAL #',I4,' *****')
     1009   FORMAT('0','***** ONLY 250 HITS WERE STORED FOR TARGET',
         X ' #',I3,' DURING TRIAL #',I4,' *****')
           END
           SUBROUTINE PLOTHT(NH,NR,LTH,WID)
           COMMON /HIT3/ XN(250),YN(250),NZ(250)
           DIMENSION ICOL(130)
           DATA IBK / 1H /,IX/ 1H* /,IY/ 1H+ /,IS/ 1H- /,IE/ 1H' /
C          THIS ROUTINE PLOTS THE IMPACT POINTS (BUT NOT CRATERS) FOR
C          ALL HITS THAT HAVE BEEN STORED FOR A RUNWAY/TAXIWAY.  IT
C          WILL PLOT ALL HITS THAT AFFECT RUNWAY OPERATION UP TO 50 'FEET'
C          OF EITHER SIDE OF (UP TO) A 300 'FOOT' RUNWAY.  RUNWAY LENGTH
C          IS LIMITED TO 13000 'FEET'.
```

```
      IWID = WID/10. + 5.
      LEN = LTH/100. + 1.
      IF (LEN .GT. 129)  LEN = 129
      LU = LEN/10.
      LI = 10*LU + 1
      LU = LU + 1
      DO 40   J = 1,40
              I = 41-J
      DO 10   N = 1,129
   10 ICOL(N) = IBK
      ICOL(1) = IE
      ICOL(LEN) = IE
      IF ((I .NE. 5) .AND. (I .NE. IWID))   GO TO 14
      DO 11   NS = 1,LEN
   11 ICOL(NS) = IS
      DO 12   NS = 1,LI,10
   12 ICOL(NS) = IE
   14 CONTINUE
      DO 20   N = 1,NH
      NY = YN(N)/10. + 5.
      IF (NY .NE. I)   GO TO 20
      NX = XN(N)/100. + 1.
      IF ((NX .LT. 1) .OR. (NX .GT. 129))   GO TO 20
      ICOL(NX) = IX
      IF (NZ(N) .EQ. 1)   ICOL(NX) = IY
   20 CONTINUE
      Y = I/5.
      LY=Y
      IF ((Y-LY) .NE. 0.0)   GO TO 30
      LYY = 5*LY - 5
      WRITE (6,101) LYY,(ICOL(M),M=1,129)
      GO TO 40
   30 WRITE (6,102)     (ICOL(M),M=1,129)
   40 CONTINUE
      WRITE (6,103)   ( I, I=1,12), NR
      RETURN
  101 FORMAT (' ',I2,129A1)
  102 FORMAT (' ',2X,129A1)
  103 FORMAT (' ',' 0', 12(8X,I2)/,
     X 40X,'   TENS BY THOUSANDS OF LENGTH UNITS ',/,/,
     X 40X,'   IMPACT POINTS ON RUNWAY NUMBER ',I2,/,
     X 37X,' (* = POINT IMPACT WPN    + = CBU CENTROID)')
      END
      SUBROUTINE JMEMO(NJMEM,D)
      COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,MCW,MCL,MODE,NPRINT,NAM,NST,MTT
      DIMENSION E(9), D(11)
C     THIS SUBROUTINE PROVIDES THE USER 'METHOD 0' AS OUTLINED IN THE
C     8-74 WANG 700 USERS'S MANUAL FOR JMEM OPEN-END METHODS.
C     THIS TRAJECTORY PROGRAM PERMITS AIDA USERS TO PRESCRIBE THE
C     ATTACK DATA AS IN JMEM.  THE SUBROUTINE LOGIC IS TAKEN DIRECTLY FROM
C     THE REFERENCED PUBLICATION AND USES NOTATION CLOSELY PARALLELING
C     THE ORIGINAL.  ONLY THE 'PATTERN RADIUS' COMPUTATION (USED WITH
C     ROCKEYE) IS OMITTED.
      NJMEM = NJMEM + 1
      NCNT = 0
      IF (NJMEM .EQ. 1)   WRITE (6,101)
      READ (5,102)  (E(I),I=1,9)
      WRITE (6,103)  NA, (D(I),I=1,6)
      WRITE (6,104)      (E(I),I=1,9)
      DO 20   I = 7,9
   20 D(I) = D(I+1)
```

```
            V1 = 1.688*E(1)
             V = V1
            TH = E(3)/57.3
            VX = V*COS(TH)
            VY = -V*SIN(TH)
            TR = (D(7) - 1.)*E(8) + (E(5)-1.)*E(9)
            DX = TR*VX/2.
            Y1 = E(2) - VY*TR
            YI = (Y1 + E(2))/2.
            Y  = YI
            TF = 0.0
            VF = 0.0
            THF=0.0
            IF (E(7) .GT. 500.)   GO TO 1
            YT = 0.0
            IF (E(7) .EQ. 0.0)   GO TO 2
            TPD=E(7)
            GO TO 3
      1     YT = E(7)
      2     TPD= 99.
      3     CONTINUE
            DG = 32.17/(E(4)*E(4))
      7     CONTINUE
            IF (KTEST .GT. 5)   WRITE (6,106) TF,Y,V
            NCNT = NCNT + 1
            IF (NCNT .GT. 2000)   GO TO 30
            NCNT2 = 0
            C = V/220. - 3.
            IF (C .GE. 3.)   C = 3.
            IF (C .LT. 1.)   C = 1.
           DD = C*DG*EXP(-Y/31000.)
            DT= 10./(DG*V*V)
            IF (DT .GT. 0.5)   DT = 0.5
            GO TO 13
      5     CONTINUE
            TF = TFO
            VY = VYO
            Y  = YO
            IF (KTEST .GT. 5)   WRITE(6,107)  DT, VY
            NCNT2 = NCNT2 + 1
            IF (NCNT2 .GT. 100)   GO TO 30
     13     CONTINUE
            TFO = TF
            TF  = TF + DT
            IF (TF .LT. TPD)   GO TO 4
            TF = TPD
            DT = TF - TFO
      4     CONTINUE
            VYO = VY
           VY  = VY*(1.-V*DT*DD) - 32.17*DT
            YO  = Y
            Y     Y + DT*(VY+VYO)/2.
            IF ((Y - YT) .GE. -1.)   GO TO 6
            ZZ = VYO*V*DD + 32.17
             Z = VYO*VYO + 2.0*(YO-YT)*ZZ
            DT = (VYO + Z**(.5))/ZZ
            GO TO 5
      6     CONTINUE
            VXO = VX
            VX  = VX*(1.-V*DT*DD)
            DX = DX + DT*(VXO+VX)/2.
```

```
          V = (VX*VX + VY*VY)**(.5)
          IF (TF .EQ. TPD)   GO TO 12
          IF (E(7) .GT. 500.)   GO TO 11
          GO TO 10
   11     AID = ABS((Y-E(7)))
          IF (AID .LT. 1.)   GO TO 12
          IF (KTEST .GE. 5)   WRITE (6,109) AID, TF
   10     CONTINUE
          IF (KTEST .GE. 5)   WRITE(6,108) Y
          IF (Y .GE. 1.)   GO TO 7
          GO TO 15
   12     CONTINUE
          Z = -VY/V
          ZZ= (1.-Z*Z)**(.5)
          THF=ATAN(Z/ZZ)
          DG = 32.17/(E(5)*E(5))
          TPD= 99.
          YT = 0.0
          IF (E(7) .GE. 500.)   E(7) = 0.0
          GO TO 10
   15     Z  = -VY/V
          ZZ = (1.-Z*Z)**(.5)
          AI = ATAN(Z/ZZ)
          SR = (DX*DX + Y1*Y1)**(.5)
          IF (D(5) .EQ. 0.0)   GO TO 8
          BP = D(4)/1000.
          CP = D(5)/1000.
          GO TO 9
    8     BP = 0.000573*D(4)
          CP = BP
    9     CONTINUE
          U  = (E(6)*E(6)*TF*TF)
          D(4)=(U+(SR*SR*BP/Y1)**2.)**(.5)
          D(5)=(U+CP*CP*SR*SR)**(.5)
          SRT = SR/1000.
          D(8) = V1*TF*SIN(AI-TH)/SIN(AI)
          DISP = D(6)
          D(10) = 0.674*DISP*SRT
          D(6)  = D(10)/Z
          AID = 57.3*AI
          THFD= 57.3*THF
          WRITE(6,105) TF, SR, AID, THFD
          RETURN
   30     WRITE (6,110)
          STOP
  101     FORMAT('1',20X,'JMEM FORMAT ATTACK DATA'//' ',
         X'ATT NO.      HDG        X-MPI      Y-MPI     CEP(REP)     (DEP)      DIS
         XP'/' ',
         X'              SPEED     ALTITUDE      DIVE      TERM1       TERM2     WIN
         XD     TD/HF          T            TD',/,/)
  102     FORMAT( 6X, 6F6.0, 3F6.3)
  103     FORMAT ('0',I6,7F10.0)
  104     FORMAT (' ',6X, 6F10.0, 3F10.3)
  105     FORMAT(' ',30X,'TF ',F5.2,' SEC    SR ',F7.0,' FEET    ',
         X 'IMPACT ANGLE ',F5.2,' DEG   (FUZING ANGLE ',F5.2,' DEG)')
  106     FORMAT(' ',' **  ',F6.3,' SEC  ',F8.0,' FEET  ',F8.1,' FT/SEC')
  107     FORMAT(' ',10X,' DT ',F10.4,'     VERT VEL',F10.3)
  108     FORMAT(' ','ALT ',F10.2)
  109     FORMAT(' ','ALT DIFF',F10.2,'  TF ',F10.2)
  110     FORMAT('0'/' ','LOOPING IN JMEMO.  CHECK INPUTS AND/OR TEST',
         X ' WITH KTEST = 6.0')
```

```
      END
      SUBROUTINE EXPHIT
      COMMON /ARRAYS/ TGT(250,13), ATT(50,11),AMD(10,20,2),TD(250,2),
     XIZONE(50,2),NHIT(250),MHIT(20),HIT(20,3,25),NRW(5),HITR(5,3,250)
     X ,P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNREL(10),NCBU(250)
      COMMON/INT/NT,NA,ND,NTM,KTEST,MCR,MCW,MCL,MODE,NPRINT,NAM,NST,MTT
      COMMON / HITDN / NHITD, NREDO
C     THIS ROUTINE ESTIMATES THE EXPECTED NUMBERS OF HITS FOR ATTACKS WITH
C     POINT-IMPACT WEAPONS ONLY.
C
C
C
      NCYCLE = 0
      PI = 3.14159
      DO 200   I = 1,NA
      NW = ATT(I,9)
      IF (AMD(NW,1,1) .LT. 0.0)   GO TO 240
      REL = WPNREL(NW)
      NFLAG = 0
      NS = ATT(I,7)
      LS = ATT(I,8)
      PHI= ATT(I,1)/57.3
      S  = SIN(PHI)
      C  = COS(PHI)
      IF ((ATT(I,1) .EQ. 0.) .OR. (ATT(I,1) .EQ. 180.))  GO TO 10
      CT = C/S
      GO TO 20
   10 NFLAG = 1
   20 CONTINUE
      NP = 10
      DLS = LS/9.
      IF (LS .GT. 500)   GO TO 30
      NP = 5
      DLS = LS/4.
      IF (LS .GT. 50)   GO TO 30
      NP = 2
      DLS = LS
   30 CONTINUE
       X = ATT(I,2) - S*LS/2.
       Y = ATT(I,3) - C*LS/2.
      SIGRS = 2.200*ATT(I,4)*ATT(I,4) + ATT(I,6)*ATT(I,6)
      SIGDS = 2.200*ATT(I,5)*ATT(I,5) + ATT(I,10)*ATT(I,10)
      TSRS = 2. *SIGRS
      TSDS = 2. *SIGDS
      SIGR = SIGRS**(.5)
      SIGD = SIGDS**(.5)
      F  = 1. / (SIGR*SIGD*6.2832)
      IF (KTEST .GT. 1)   WRITE(6,1008)I,LS,NP,DLS,X,Y,SIGR,
     X SIGD, F
      NFLAG2 = 0
      DO 180   L = 1,NT
      IF (TGT(L,10) .EQ. 21.)   GO TO 180
      NFLAG2 = 1
      DEN = 0.0
      TL1 = TGT(L,12)
      TL2 = TGT(L,13)
      TH = TGT(L,9)/57.3
      CHI = TH - PHI
      SC = SIN(CHI)
      CC = COS(CHI)
      S1 = (SIGRS*CC*CC + SIGDS*SC*SC)**(.5)
```

```
      S2 = (SIGRS*SC*SC + SIGDS*CC*CC)**(.5)
      N1 = (4.*TL1/S1) + 1
      N2 = (4.*TL2/S2) + 1
      IF (KTEST .GT. 1)   WRITE(6,1009) N1, N2, S1, S2
      DO 90   K = 1,7,2
      XT = TGT(L,K)
      YT = TGT(L,K+1)
      XX = X - XT
      IF (NFLAG .EQ. 1)   GO TO 40
      YY = YT - Y + CT*XX
       D = -YY*S
       R =  YY*C - XX/S
      GO TO 50
   40 D = XX
      R = (YT - Y)*C
   50 AID1 = D*D/TSDS
      IF ((TL1+TL2) .GT. 200.)   GO TO 60
      IF (AID1 .GT. 10.)   GO TO 180
   60 IF (AID1 .GT. 10.)   GO TO 90
      DD = SEXP(-AID1)
      DO 80   M = 1,NP
      AID2 = R*R/TSRS
      IF (AID2 .GT. 10.)   GO TO 70
      DR = SEXP(-AID2)
      DEN = DEN + DR*DD
      NCYCLE = NCYCLE + 1
      IF (KTEST .GT. 2) WRITE(6,1005)I,L,K,XT,YT,D,R,DD,DR,DEN
   70 R = R - DLS
   80 CONTINUE
   90 CONTINUE
C     IF THE TARGET DIMENSIONS ARE SMALL (I.E. LESS THAN ONE-QUARTER
C     THE PROJECTION OF SIGMA PARALLEL TO THE TARGET EDGE) THE HIT
C     DENSITY IS TAKEN AS THE AVERAGE OF THE VALUES AT THE FOUR CORNERS.
C     IF IT IS LARGER, A GRID OF INTERNAL POINTS IS ESTABLISHED AND
C     THE HIT DENSITY IS TAKEN AS THE AVERAGE OVER THE CORNERS AND THE
C     INTERNAL POINTS.
      IF ((N1 + N2) .GT. 2)   GO TO 100
      DEN = DEN/(4.*NP)
      GO TO 160
  100 CONTINUE
      STG = SIN(TH)
      CTG = COS(TH)
      DIL1 = TL1/(N1+1)
      DIL2 = TL2/(N2+1)
      DO 150   M = 1,N1
      DO 150   N = 1,N2
      XT = TGT(L,1) + M*DIL1*STG + N*DIL2*CTG
      YT = TGT(L,2) + M*DIL1*CTG - N*DIL2*STG
      XX = X - XT
      IF (NFLAG .EQ. 1)   GO TO 110
      YY = YT - Y + CT*XX
      D = -YY*S
      R =  YY*C - XX/S
      GO TO 120
  110 D = XX
      R = (YT - Y)*C
  120 AID1 = D*D/TSDS
       IF (AID1 .GT. 10.)   GO TO 150
      DD = SEXP(-AID1)
      DO 140   K = 1,NP
      AID2 = R*R/TSRS
```

```
              IF (AID2 .GT. 10.)    GO TO 130
              DR = SEXP(-AID2)
              DEN = DEN + DR*DD
              NCYCLE = NCYCLE + 1
              IF (KTEST .GT. 2) WRITE(6,1006)I,L,XT,YT,D R,DD,DR,DEN
        130   R = R - DLS
        140   CONTINUE
        150   CONTINUE
                DEN = DEN/ ((4+N1*N2)*NP)
        160   CONTINUE
              DEN1 = DEN
              DEN = NS*F*REL*ATT(I,11)*DEN1
              IF (KTEST .GT. 1)    WRITE (6,1007) I,L,DEN1,DEN
              EMD = AMD(NW,TGT(L,10),1)
              DL1 = TL1 + 2*EMD
              DL2 = TL2 + 2*EMD
              EHIT1 = DEN*(DL1*DL2 - .8584*EMD*EMD)
              COV(L) = COV(L) + EHIT1
        180   CONTINUE
              IF (NFLAG2 .EQ. 0)    GO TO 230
        200   CONTINUE
              WRITE(6,1000) NCYCLE
              WRITE (6,1002)
              DO 220    M = 1,MTT
              NN = 0
              TCOV = 0.0
              BLDGHT = 0.0
              DO 210    L = 1,NT
              IF (TGT(L,10) .NE. M)    GO TO 210
              NN = NN + 1
              TCOV = TCOV + COV(L)
              BLDGHT = BLDGHT + (1. - EXP(-COV(L)))
              IF (NN .EQ. 1)    WRITE (6,1001) M
              WRITE (6,1003) L,COV(L),NAME(L,1),NAME(L,2)
        210   CONTINUE
              IF (NN .EQ. 0)    GO TO 220
              BLDGHT = BLDGHT/NN
              WRITE (6,1010)   TCOV, BLDGHT
        220   CONTINUE
        230   CONTINUE
              RETURN
        240   WRITE (6, 1004)
              STOP
       1000    FORMAT('1', 25X, 'CYCLES', I7//)
       1001    FORMAT ('0',10X,'** TARGET TYPE #',I3,' **',/)
       1002    FORMAT ('0',10X,'TARGET      HITS',/,10X,'  NO        EXPECTED')
       1003    FORMAT (' ',10X,I4,6X,F6.3,6X,2A4)
       1004    FORMAT ('1','    COMPUTATION STOPPED - CBU WEAPONS ARE NOT',
           X    ' PERMITTED WITH SUBROUTINE EXPHIT')
       1005    FORMAT(' ','CD    ATT ',I3,'    TGT ',I3,'    COR ',I2,
            X 4F6.0, 3E12.4 )
       1006    FORMAT (' ','IP    ATT ',I3,'    TGT ',I3,4F6.0,3E12.5)
       1007    FORMAT(' ','         ATT ',I3,'    TGT ',I3,'    AVG DEN ',
           X     F8.5,'    NOR DEN ',F14.10,//)
       1008    FORMAT (' ','ATT ',I3,2I8, F8.2, 4F8.1, E12.5)
       1009    FORMAT(' ','N1    ',I4,'      N2    ',I4,6X,2F10.1)
       1010    FORMAT(' ',20X,'-------',/,19X,F8.3,'(',F5.3,')')
              END
              FUNCTION SEXP(X)
              IF (X .LT. -0.025)    GO TO 10
              SEXP = 1.+X
```

```
      RETURN
   10 SEXP = EXP(X)
      RETURN
      END
      SUBROUTINE GRIDEN
     COMMON/ARRAYS/TGT(250,13),ATT(50,11),AMD(10,20,2),TO(250,2),
    XIZONE(50,2),NHIT(250),MHIT(20),HIT(20,3,25),NRW(5),HITR(5,3,250)
    X,P(250,3),COV(250),MTYPE(10),NAME(250,2),WPNREL(10),NCBU(250)
     COMMON/INT/NT,NA,NO,NTM,KTEST,MCR,MCW,MCL,MODE,NPRINT,NAM,NST,MTT
      DIMENSION DATA(17)
        NCYCLE = 0
        PI = 3.14159
        SRPI = 1.7724
     DO 140   L = 1,NT
        IF (TGT(L,10) .NE. 21.)   GO TO 140
        NFLAG2 = 0
        INC = 250
        IF (TGT(L,9) .GT. 0.)   INC = TGT(L,9)/16.
        NX1 = TGT(L,1)
        NY1 = TGT(L,2)
      DO 120   NYR = 1,17
      DO 120   NX  = 1,17
        NY = 18 - NYR
         DEN = 0
        XT = NX1 + (NX-1)*INC
        YT = NY1 + (NY-1)*INC
         DATA(NX) = 0.0
      DO 100   I = 1, NA
         NW = ATT(I,9)
         IF (AMD(NW,1,1) .LT. 0.)   GO TO 160
         REL = WPNREL(NW)
         NFLAG = 0
         NS = ATT(I,7)
         LS = ATT(I,8)
        PHI = ATT(I,1)/57.3
        S = SIN(PHI)
        C = COS(PHI)
        IF ((ATT(I,1) .EQ. 0.) .OR. (ATT(I,1) .EQ. 180.))   GO TO 10
        CT = C/S
        GO TO 20
   10   NFLAG = 1
   20   CONTINUE
        NP = 10
        DLS = LS/9.
        IF (LS .GT. 500)   GO TO 30
        NP = 5
        DLS = LS/4.
        IF (LS .GT. 50)   GO TO 30
        NP = 2
        DLS = LS
   30   CONTINUE
        X = ATT(I,2) -S*LS/2.
        Y = ATT(I,3) -C*LS/2.
        SIGRS = 2.200*ATT(I,4)*ATT(I,4) + ATT(I,6)*ATT(I,6)
        SIGDS = 2.200*ATT(I,5)*ATT(I,5) + ATT(I,10)*ATT(I,10)
        TSRS = 2.0 * SIGRS
        TSDS = 2.0 * SIGDS
        SIGR = SIGRS**(.5)
        SIGD = SIGDS**(.5)
        F = 1. / (SIGR*SIGD*6.2832)
        XX = X - XT
```

```
              IF (NFLAG .EQ. 1)   GO TO 40
              YY = YT - Y + CT*XX
               D = -YY*S
               R =  YY*C - XX/S
              GO TO 50
   40          D = XX
               R = (YT - Y)*C
   50         AID1 = D*D/TSDS
              IF (AID1 .GT. 10.)   GO TO 90
              DD = SEXP(-AID1)
              NCYCLE = NCYCLE + 1
            DO 80   M = 1, NP
              AID2 = R*R/TSRS
              IF (AID2 .GT. 10.)   GO TO 70
              DR = SEXP(-AID2)
              DEN = DEN + DR*DD
              NCYCLE = NCYCLE + 1
              IF (KTEST .GT. 2)   WRITE (1005) XT,YT,D,R,DD,DR,DEN
   70         R = R - DLS
   80       CONTINUE
   90       CONTINUE
              DEN = 10000*NSAF*REL*ATT(I,11)*DEN/NP
              DATA(NX) = DATA(NX) + DEN
  100       CONTINUE
              IF (NFLAG2 .EQ. 1)   GO TO 110
              NFLAG2 = 1
              WRITE (6, 1001)
  110       CONTINUE
              IF (NX .LT. 17)    GO TO 120
              NYT = YT
              IF (NY .LT. 17)    GO TO 115
              NX2 = NX1 + 16*INC
              WRITE (6,1002)    (NLOC,  NLOC=NX1,NX2,INC)
  115         WRITE (6,1003)    NYT,  (DATA(I), I=1,17)
  120       CONTINUE
  140       CONTINUE
               WRITE (6,1004)   NCYCLE
            RETURN
  160       CONTINUE
              WRITE(5,1006)
              STOP
 1001         FORMAT('1',20X,'EXPECTED HIT DENSITY PER 10000 SQ FT',///)
 1002         FORMAT(' ',//,30X,' X - LOCATION',/,' ',10X,17I7,/,
     X   '  Y LOC    ')
 1003         FORMAT(' ',/,' ', I6, 4X, 17F7.3 )
 1004         FORMAT(' ',///,20X,'CYCLES IN GRIDEN ',I8)
 1006         FORMAT(' ',///,'  COMPUTATION STOPPED - CBU WEAPONS ARE',
     X   ' NOT PERMITTED WITH SUBROUTINE GRIDEN')
 1005         FORMAT(' ',4F6.0,3E12.4)
            END
```